



**UNIVERSIDAD NACIONAL DE INGENIERIA**  
**FACULTAD DE CIENCIAS Y SISTEMAS**

Trabajo Monográfico para optar al título de  
Ingeniero de Sistemas

**Tema:**

Desarrollo de un sistema de información para planificación de horarios  
académicos en la Universidad Nacional de Ingeniería.

**Autores:**

Br. Sergio Alexander Velásquez Rugama 2008-21047

Br. Justo Leonel Rodríguez Valdivia 2008-21133

**Tutor:**

Ing. José Manuel Poveda

Managua, 17 de marzo de 2017

## **Dedicatoria**

Gracias a Dios por sus bendiciones,  
y permitir que hayamos alcanzado  
la meta propuesta.

A cada uno de nuestros maestros,  
por sus enseñanzas y consejos.

A la comunidad online,  
que comparte sus conocimientos  
de forma desinteresada.

## **Resumen**

El desarrollo de esta investigación en la Universidad Nacional de Ingeniería, Recinto Universitario Augusto C. Sandino ha sido con la finalidad de poner a disposición de las autoridades de la sede, una herramienta que sirva de apoyo en el proceso de formación de horarios académicos, para las diferentes modalidades que esta casa de estudios ofrece.

En primera instancia se han identificado las dificultades que presenta el procedimiento actual. Con la ayuda de los diferentes involucrados se ha determinado cuales son los requerimientos funcionales y no funcionales que el sistema propuesto deberá cumplir para tener un efecto significativo en el proceso de creación de horarios académicos.

Se ha verificado la factibilidad del proyecto mediante un estudio de factibilidad técnica, económica y organizacional, esto con el fin de alcanzar el éxito en la implementación del sistema propuesto.

El software fue creado siguiendo una metodología de desarrollo iterativa llamada Scrum, al final de la cual se obtuvo un producto que opera bajo la arquitectura cliente-servidor, conectado al gestor de base de datos MySQL y funcionando en ordenadores con sistemas Windows. El resultado final ha sido documentado con diagramas UML que representen tanto la parte estática como dinámica del sistema.

Por último, se ha desarrollado un plan de pruebas para validar el correcto funcionamiento del sistema y garantizar que cumple con los requerimientos planteados por los interesados.

Cabe señalar que el producto obtenido de esta investigación es extensible a otros recintos de la Universidad Nacional de Ingeniería, incluso, a otras instituciones de educación superior.

## Tabla de contenido

I.	Introducción .....	1
II.	Objetivos.....	2
A.	General .....	2
B.	Específicos.....	2
III.	Justificación.....	3
IV.	Marco teórico.....	4
A.	Generalidades del software .....	4
B.	Lenguaje de modelado unificado .....	6
C.	Metodología ágil de desarrollo.....	8
D.	Planificación académica .....	14
E.	Herramientas.....	14
Capítulo 1.	Enfoque sistémico .....	16
1.1	Datos generales de la institución .....	16
1.1.1	Misión .....	16
1.1.2	Visión.....	16
1.1.3	Objetivo .....	16
1.1.4	Estructura organizacional.....	17
1.2	Definición del problema .....	17
1.3	Árbol de problemas .....	19
1.1	Árbol de objetivos.....	20
1.4	Consecuencias negativas.....	20
1.5	Consecuencias positivas .....	21
1.6	Grupos de interés.....	22

1.7	Alternativas de solución.....	22
1.8	Evaluación de soluciones .....	23
Capítulo 2.	Estudios de factibilidad .....	24
2.1	Factibilidad técnica .....	24
2.1.1	Red local.....	24
2.1.2	Equipos de cómputo .....	26
2.2	Factibilidad económica .....	28
2.2.1	Estimación de costo del sistema .....	29
2.3	Factibilidad operativa .....	44
2.3.1	Capital humano.....	44
2.3.2	Complejidad de adopción.....	46
Capítulo 3.	Análisis de requerimientos.....	48
3.1	Descripción del proceso .....	48
3.2	Administración del sistema .....	48
3.2.1	Definición de parámetros .....	48
3.2.2	Creación de cuentas y permisos .....	49
3.3	Formación de horarios.....	49
3.4	Generación de informes .....	50
3.5	Resumen de requerimientos .....	51
3.6	Detalle de requerimientos.....	52
3.7	Requerimientos no funcionales .....	58
3.7.1	Portabilidad.....	58
3.7.2	Flexibilidad.....	58
3.7.3	Disponibilidad .....	58

3.7.4	Madurez.....	59
3.7.5	Concurrencia .....	59
3.7.6	Interface de usuario .....	59
3.7.7	Desempeño .....	59
Capítulo 4.	Proceso de iteraciones Scrum .....	60
4.1	Historial de revisiones .....	60
4.2	Roles.....	61
4.3	Product Backlog .....	61
4.4	Gráfica de producto.....	62
4.5	Sprint #1 Gestión de infraestructura .....	63
4.5.1	Sprint Planning Meeting.....	63
4.5.2	Sprint Review Meeting .....	63
4.5.3	Burn down chart.....	63
4.5.4	Sprint Backlog.....	64
4.6	Sprint #2 Gestión de docentes .....	65
4.6.1	Sprint Planning Meeting.....	65
4.6.2	Sprint Review Meeting .....	65
4.6.3	Burn down chart.....	65
4.6.4	Sprint Backlog.....	66
4.7	Sprint #3 Gestión de tiempo .....	67
4.7.1	Sprint Planning Meeting.....	67
4.7.2	Sprint Review Meeting .....	67
4.7.3	Burn down chart.....	67
4.7.4	Sprint Backlog.....	68

4.8	Sprint #4 Gestión de pensum .....	69
4.8.1	Sprint Planning Meeting .....	69
4.8.2	Sprint Review Meeting .....	69
4.8.3	Burn down chart .....	69
4.8.4	Sprint Backlog .....	70
4.9	Sprint #5 Gestión de horarios .....	71
4.9.1	Sprint Planning Meeting .....	71
4.9.2	Sprint Review Meeting .....	71
4.9.3	Burn down chart .....	71
4.9.4	Sprint Backlog .....	72
4.10	Sprint #6 Gestión de informes .....	73
4.10.1	Sprint Planning Meeting .....	73
4.10.2	Sprint Review Meeting .....	73
4.10.3	Burn down chart .....	73
4.10.4	Sprint Backlog .....	74
4.11	Sprint #7 Gestión de usuarios .....	75
4.11.1	Sprint Planning Meeting .....	75
4.11.2	Sprint Review Meeting .....	75
4.11.3	Burn down chart .....	75
4.11.4	Sprint Backlog .....	76
4.12	Sprint #8 Integración general .....	77
4.12.1	Sprint Planning Meeting .....	77
4.12.2	Sprint Review Meeting .....	77
4.12.3	Burn down chart .....	77

4.12.4	Sprint Backlog .....	78
Capítulo 5.	Diseño UML.....	79
5.1	Diagramas de clases .....	79
5.1.1	Base .....	79
5.1.2	Infraestructura.....	79
5.1.3	Docentes .....	80
5.1.4	Horarios .....	80
5.1.5	Pensum .....	81
5.1.6	Tiempos.....	81
5.1.7	Usuarios .....	82
5.1.8	Reportes .....	82
5.2	Diagramas de secuencia .....	83
5.2.1	Inicio de sesion .....	83
5.2.2	Navegar .....	84
5.2.3	Crear nuevo elemento .....	84
5.2.4	Editar elemento existente.....	85
5.2.5	Cambiar estado .....	85
5.2.6	Filtrar horarios.....	86
5.2.7	Validar choques .....	86
5.2.8	Imprimir reportes.....	87
5.3	Diagrama de distribución.....	88
5.4	Diagrama de componentes .....	88
Capítulo 6.	Modelo de base de datos .....	89
6.1	Modelo de infraestructura .....	89



6.2	Modelo de docentes .....	89
6.3	Modelo de tiempos .....	90
6.4	Modelo de pensum .....	90
6.5	Modelo de horarios.....	91
6.6	Modelo de usuarios .....	91
6.7	Modelo de reportes .....	92
6.8	Modelo global .....	93
6.9	Diccionario de datos .....	94
Capítulo 7.	Presentación del sistema.....	107
7.1	Inicio de sesión .....	107
7.2	Pantalla principal .....	107
7.3	Control de usuarios .....	108
7.4	Crear registros .....	111
7.5	Formar horarios.....	112
7.6	Listar registros.....	113
7.7	Filtrar horarios .....	114
7.8	Reportes.....	114
7.8.1	Análisis .....	116
Capítulo 8.	Pruebas en entorno controlado.....	118
8.1	Introducción.....	118
8.2	Objetivos del plan .....	118
8.3	Documentos relacionados .....	118
8.4	Alcance de las pruebas .....	119
8.5	Entorno y configuración de las pruebas.....	120

8.6	Criterios de aprobación o rechazo .....	120
8.6.1	Tipos de errores.....	120
8.7	Estrategia de pruebas .....	121
8.8	Escenario de las pruebas .....	121
V.	Conclusiones .....	123
VI.	Recomendaciones .....	124
VII.	Bibliografía .....	125
VIII.	Anexos .....	127
A.	Entrevista a coordinadores de carrera.....	127

## **I. Introducción**

La Universidad Nacional de Ingeniería (UNI) fue creada el 7 de febrero de 1983, con el objetivo de formar profesionales en el ramo de las ingenierías y la arquitectura, con los más altos niveles científicos, tecnológicos y humanistas.

En el año 2004 la UNI decide ejecutar un proyecto de expansión abriendo un recinto en el departamento de Estelí, actualmente llamado: Recinto Universitario Augusto C. Sandino (UNI-RUACS). Los departamentos de la UNI-RUACS deben compartir recursos, tales como: aulas, medios audiovisuales y profesores, esto sumado al crecimiento que la sede ha experimentado desde su fundación, genera un incremento en la complejidad de la planificación horaria.

Se propone el desarrollo de un sistema informático para gestionar la creación de horarios, y así reducir la complejidad administrativa que representa esta tarea. Dicho sistema gestionará los docentes, carreras, aulas, turnos, períodos, medios audiovisuales, asignaturas y grupos, estará disponible para trabajo en red, y funcionará bajo una arquitectura cliente-servidor.

Este trabajo consiste en la documentación de dicha propuesta, se abordan los aspectos de: ingeniería de requerimientos, donde se realiza un estudio de las necesidades del usuario, para determinar las características y funciones que deberá incorporar el sistema; incluye estudios técnicos, económicos, y operacionales que tienen como fin comprobar la factibilidad del proyecto; análisis y diseño del sistema, donde se abstraen los requerimientos (tanto funcionales, como no funcionales) con el fin de obtener un modelo que represente las partes, interacciones, relaciones, flujos de información, restricciones, y elementos involucrado en el sistema; desarrollo mediante la metodología Scrum, con el fin de obtener un producto que cumpla con los requerimientos y diseños planteados; pruebas en entorno controlado, donde se verificará el correcto funcionamiento del sistema.

## **II. Objetivos**

### **A. General**

- Desarrollar un sistema informático para la gestión de horarios académicos en la Universidad Nacional de Ingeniería, Recinto Universitario Augusto C. Sandino.

### **B. Específicos**

- Realizar un análisis de la situación problemática a través del enfoque sistémico.
- Determinar los requerimientos del sistema a desarrollar, a través de la aplicación de entrevistas a los involucrados para definir características y funciones.
- Realizar análisis de factibilidad técnica, económica y operativa del sistema propuesto.
- Diseñar el sistema aplicando técnicas de ingeniería de software orientado a objetos.
- Validar el funcionamiento del sistema mediante la realización de pruebas en entorno controlado y entrevistas a los actores.

### **III. Justificación**

El presente trabajo surge ante la necesidad de crear un sistema informático, que mejore y simplifique los métodos de planificación horaria utilizados en la Universidad Nacional de Ingeniería, Recinto Universitario Augusto C. Sandino.

El desarrollo de un sistema informático para planificación de horarios académicos aporta un enorme beneficio para la universidad, puesto sus funciones y características a la medida, facilitarán el proceso de planificación horaria, por ejemplo: habilitar el trabajo en red, lo que permite una mayor integración por parte de los encargados de la elaboración de la carga horaria; proveer de herramientas y mecanismos que simplifiquen el monitoreo de conflictos entre los diferentes horarios académicos, de modo que no se admitan los choques de asignaturas para un mismo docente o grupo; facilitar la visualización e impresión de reportes, que muestran al usuario la carga horaria de forma limpia y rápida; realizar cambios de forma sencilla y segura en los datos y horarios almacenados previamente, lo que aporta flexibilidad al trabajo realizado; trabajar con un sistema de base de datos, lo que haga innecesario el llenado constante de información redundante; optimizar los tiempos de registros de datos, obteniendo un incremento en la productividad de los empleados y demás involucrados.

Además de lo anteriormente mencionado, el uso de un software a la medida que sea propiedad de la universidad, aumenta el prestigio y mejora la imagen de la casa de estudio. La Universidad Nacional de Ingeniería es líder en ciencia y tecnología, por lo que no puede utilizar software sin licencias originales, versiones demo de aplicaciones enlatadas, o sistemas que no funcionen según las necesidades y requerimientos específicos de la institución.

## IV. Marco teórico

### A. Generalidades del software

Un **sistema de información (SI)** es un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización. Además de apoyar la toma de decisiones, la coordinación y el control, los sistemas de información también pueden ayudar a los gerentes y trabajadores del conocimiento a analizar problemas, visualizar temas complejos y crear nuevos productos. Dichos componentes formarán parte de alguna de las siguientes categorías:

- Personas
- Datos
- Actividades o técnicas de trabajo
- Recursos materiales en general (generalmente recursos informáticos y de comunicación, aunque no necesariamente).

Todos estos componentes interactúan para procesar los datos (incluidos los procesos manuales y automáticos) y dan lugar a información más elaborada, que se distribuye de la manera más adecuada posible en una determinada organización, en función de sus objetivos. (Laudon & Laudon, 2012)

Es común utilizar software como recurso informático dentro de los sistemas de información, este cumple el papel de equipamiento lógico de una computadora digital; comprende el conjunto de los componentes necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos, que son llamados hardware.

Los componentes lógicos incluyen, entre muchos otros, las aplicaciones informáticas; tales como el procesador de textos, que permite al usuario realizar todas las tareas concernientes a la edición de textos; el software de sistema, tal

como el sistema operativo, que, básicamente, permite al resto de los programas funcionar adecuadamente, facilitando también la interacción entre los componentes físicos y el resto de las aplicaciones, y proporcionando una interfaz para el usuario. (Real Academia Española, 2005)

Existen dos grandes categorías en las que se pueden clasificar el software:

- El **Software a medida** es un producto de Ingeniería de Software desarrollado especialmente para un cliente en particular, teniendo en cuenta sus propios requerimientos y la política de su empresa u organización.
- El **Software de uso general** (enlatado) está orientado al mercado masivo con clientes que comparten las mismas necesidades. El software enlatado por lo general impide que se realicen modificaciones de fondo para adaptarse mejor a las necesidades del cliente.

El software también puede ser clasificado según su finalidad, por ejemplo: el enfocado en la planeación de horario académicos, es aquel que se destina a la creación y seguimiento de horarios de clase, se caracterizan por brindar control sobre, asignaturas, aulas, docentes, y horas clase, es usado comúnmente por universidades, y en menor medida por centros de educación básica. En este contexto debe entenderse la planificación como el proceso metódico, diseñado para obtener un objetivo determinado, dicho de una forma más precisa: "La planificación es un proceso de toma de decisiones para alcanzar un futuro deseado, teniendo en cuenta la situación actual y los factores internos y externos que pueden influir en el logro de los objetivos". (Jimenez, 2013)

## B. Lenguaje de modelado unificado

Con el fin de representar el funcionamiento del software de forma gráfica, descriptiva y legible, se han creado lenguajes de modelado, uno de ellos es el **Unificado de Modelado** (LUM o **UML**, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

Es importante resaltar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software, pero no especifica en sí mismo qué metodología o proceso usar. (Schmuller, 2001)

UML contempla diferentes diagramas, entre ellos:

- Diagrama de casos de uso
- Diagrama de clases
- Diagrama de secuencia
- Diagrama de estados
- Diagrama de actividad

Los **casos de uso**, son una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores. En el contexto de



ingeniería del software, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Una relación es una conexión entre los elementos del modelo, por ejemplo, la especialización y la generalización son relaciones. Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona a eventos que se producen en su ámbito o en él mismo.

Los casos de uso son los diagramas más comunes para la captura de requisitos funcionales, especialmente con el desarrollo del paradigma de la programación orientada a objetos, donde se originaron, si bien puede utilizarse con resultados igualmente satisfactorios con otros paradigmas de programación. (Schmuller, 2001)

Uno de los elementos principales que forman los casos de uso son los actores, se le llama actor a toda entidad externa al sistema que guarda una relación con éste y que le demanda una funcionalidad. Esto incluye a los operadores humanos, pero también incluye a todos los sistemas externos, además de entidades abstractas, como el tiempo.

Suele suceder, sin embargo, que es el sistema quien va a tener interés en el tiempo. Es frecuente encontrar que nuestros sistemas deben efectuar operaciones automáticas en determinados momentos; y siendo esto un requisito funcional obvio, resulta de interés desarrollar alguna forma de capturar dicho requisito en el modelo de caso de uso final. (Schmuller, 2001)

Un **diagrama de clases** es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la

relación entre uno y otro. (Schmuller, 2001)

Los **diagramas de estado** muestran el conjunto de estados por los cuales pasa un objeto durante su vida en una aplicación en respuesta a eventos (por ejemplo, mensajes recibidos, tiempo rebasado o errores), junto con sus respuestas y acciones. También ilustran qué eventos pueden cambiar el estado de los objetos de la clase. Normalmente contienen: estados y transiciones. Al igual que otros diagramas, en los diagramas de estado pueden aparecer notas explicativas y restricciones. (Schmuller, 2001)

Un **diagrama de secuencia** muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Mientras que el diagrama de casos de uso permite el modelado de una vista *business* del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes intercambiados entre los objetos.

Típicamente se examina la descripción de un caso de uso para determinar qué objetos son necesarios para la implementación del escenario. Si se dispone de la descripción de cada caso de uso como una secuencia de varios pasos, entonces se puede "caminar sobre" esos pasos para descubrir qué objetos son necesarios para que se puedan seguir los pasos. Un diagrama de secuencia muestra los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como flechas horizontales. (Schmuller, 2001)

En el Lenguaje de Modelado Unificado, un **diagrama de actividades** representa los flujos de trabajo paso a paso de negocio y operacionales de los componentes en un sistema. (Schmuller, 2001)

### **C. Metodología ágil de desarrollo**

El **desarrollo ágil de software** son métodos de ingeniería del software basados en el desarrollo iterativo e incremental, donde los requerimientos y soluciones evolucionan mediante la colaboración de grupos auto-organizados y

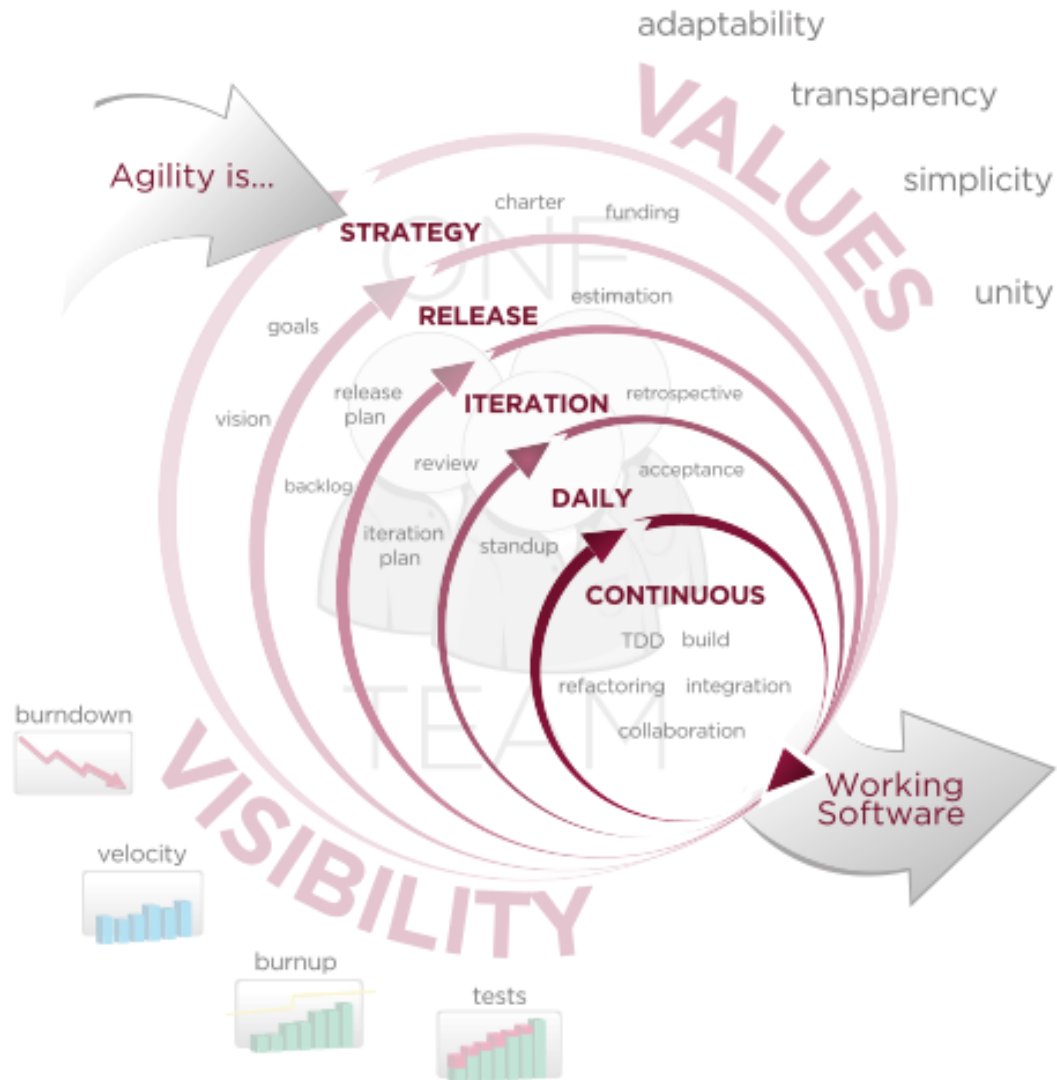
multidisciplinarios.

Existen muchos métodos de desarrollo ágil; la mayoría minimiza riesgos desarrollando software en lapsos cortos. El software desarrollado en una unidad de tiempo es llamado una iteración, la cual debe durar de una a cuatro semanas. Cada iteración del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación. Una iteración no debe agregar demasiada funcionalidad para justificar el lanzamiento del producto al mercado, pero la meta es tener una *demo* (sin errores) al final de cada iteración. Al final de cada iteración el equipo vuelve a evaluar las prioridades del proyecto. (Cockburn, 2000)

**Scrum** es una metodología ágil de desarrollo de proyectos que toma su nombre y principios de los estudios realizados sobre nuevas prácticas de producción por Hirotaka Takeuchi e Ikujiro Nonaka a mediados de los 80.

Aunque surgió como modelo en el desarrollo de productos tecnológicos, sus principios son válidos para entornos que trabajan con requisitos inestables, y necesitan agilidad: situaciones frecuentes en el desarrollo de determinados sistemas de software.

# AGILE DEVELOPMENT



## ACCELERATE DELIVERY

Figura1: Metodología de desarrollo ágil (Beck, 2001)

Scrum es una metodología ágil:

- Es un modo de desarrollo de carácter adaptable.
- Orientado a las personas antes que a los procesos.
- Emplea desarrollo ágil: iterativo e incremental.

El desarrollo se inicia desde la visión general de producto, dando detalle solo a las funcionalidades que, por ser las de mayor prioridad para el negocio, se van a desarrollar en primer lugar, y pueden llevarse a cabo en un periodo de tiempo breve (menor a 60 días).

Cada uno de los ciclos de desarrollo es una iteración (sprint) que produce un incremento terminado y operativo del producto. Estas iteraciones son la base del desarrollo ágil, y Scrum gestiona su evolución a través de reuniones breves de seguimiento en las que todo el equipo revisa el trabajo realizado desde la reunión anterior y el previsto hasta la reunión siguiente. El protocolo de Scrum para Software definido por Jeff Sutherland y Ken Schwaber prescribe que las reuniones de seguimiento del sprint (iteración) sean diarias (Palacio, 2007)

El **Sprint** es el período en el cual se lleva a cabo el trabajo en sí. Es recomendado que la duración de los sprints sea constante y definida por el equipo con base en su propia experiencia. Se puede comenzar con una duración de sprint en particular e ir ajustándolo con base en el ritmo del equipo, aunque sin relajarlo demasiado. Al final de cada sprint, el equipo deberá presentar los avances logrados, y el resultado obtenido es un producto potencialmente entregable al cliente. Asimismo, se recomienda no cambiar los objetivos del sprint o *sprint backlog* a menos que la falta de estos cambios amenace al éxito del proyecto. (Schwaber, 2010)

Dentro de la metodología Scrum se contempla la participación de diferentes entidades, entre ellas:

- Product Owner

- Scrum Master
- Equipo de trabajo

El **Product Owner** representa la voz del cliente. Se asegura de que el equipo Scrum trabaja de forma adecuada desde la perspectiva del negocio. El Product Owner escribe historias de usuario, las prioriza, y las coloca en el Product Backlog. (Schwaber, 2010)

El *Scrum* es facilitado por un **ScrumMaster**, cuyo trabajo primario es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. El *ScrumMaster* no es el líder del equipo (porque ellos se auto-organizan), sino que actúa como una protección entre el equipo y cualquier influencia que le distraiga. El ScrumMaster se asegura de que el proceso Scrum se utiliza como es debido. El ScrumMaster es el que hace que las reglas se cumplan. (Schwaber, 2010)

El **equipo** tiene la responsabilidad de entregar el producto. Un pequeño equipo de 3 a 9 personas con las habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc). (Schwaber, 2010)

Además de contemplar la participación de individuos y grupos, la metodología Scrum, utiliza varios documentos para controlar y dar seguimiento al desarrollo:

- Product backlog
- Sprint backlog
- Burn down chart

El **product backlog** es un documento de alto nivel para todo el proyecto. Contiene descripciones genéricas de todos los requerimientos, funcionalidades deseables, etc. priorizadas según su retorno sobre la inversión (ROI). Es el *qué* va a ser construido. Es abierto y cualquiera puede modificarlo. Contiene estimaciones *grosso modo*, tanto del valor para el negocio, como del esfuerzo de

desarrollo requerido. Esta estimación ayuda al *product owner* a ajustar la línea temporal y, de manera limitada, la prioridad de las diferentes tareas. Por ejemplo, si dos características tienen el mismo valor de negocio la que requiera menor tiempo de desarrollo tendrá probablemente más prioridad, debido a que su ROI será más alto. (Schwaber, 2010)

El **sprint backlog** es un documento detallado donde se describe el *cómo* el equipo va a implementar los requisitos durante el siguiente sprint. Las tareas se dividen en *horas* con ninguna tarea de duración superior a 16 horas. Si una tarea es mayor de 16 horas, deberá ser dividida en otras menores. Las tareas en el *sprint backlog* nunca son asignadas, son tomadas por los miembros del equipo del modo que les parezca oportuno. (Schwaber, 2010)

La **burn down chart** es una gráfica mostrada públicamente que mide la cantidad de requisitos en el Backlog del proyecto pendientes al comienzo de cada Sprint. Dibujando una línea que conecte los puntos de todos los Sprints completados, podremos ver el progreso del proyecto. Lo normal es que esta línea sea descendente (en casos en que todo va bien en el sentido de que los requisitos están bien definidos desde el principio y no varían nunca) hasta llegar al eje horizontal, momento en el cual el proyecto se ha terminado (no hay más requisitos pendientes de ser completados en el Backlog). Si durante el proceso se añaden nuevos requisitos la recta tendrá pendiente ascendente en determinados segmentos, y si se modifican algunos requisitos la pendiente variará o incluso valdrá cero en algunos tramos. (Schwaber, 2010)

Antes de cada sprint se realiza una reunión llamada **Sprint Planning Meeting** en la cual se define el trabajo que se realizará y se establece el Sprint Backlog. Al finalizar el sprint se realiza una reunión llamada **Sprint Review Meeting** en la que se revisa que trabajo fue realizado y que trabajo no fue realizado, los resultados se exponen ante todo el equipo.

#### D. Planificación académica

Las entidades educativas realizan planes en los que se definen los horarios de clase que deberán cumplir los diferentes grupos y docentes. Dentro de este proceso de planificación destacan elementos como:

- **Semestre:** periodo de tiempo que dura seis meses. Las universidades suelen dividir sus actividades en semestres, aunque también existen algunas que se organizan por trimestre; esto se efectúa según conveniencia, y va a variar de una entidad de educación a otra.
- **Periodo de clase:** duración estándar de las horas clases, suele estar comprendido entre 45-60min.
- **Bloque:** unión de dos o más periodos de clase consecutivos de una misma asignatura.
- **Pensum:** asignaturas que un grupo tiene cargadas en un semestre determinado, suelen ser valores fijos, o varían muy poco en el tiempo.

#### E. Herramientas

Se utilizarán diferentes herramientas con el fin de aplicar las metodologías, conceptos y alcanzar los objetivos propuestos, las más relevantes son:

- **Microsoft Visual Studio** es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión .NET 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles. (Microsoft, 2015)

Visual Studio es el IDE por excelencia para los lenguajes .NET. El lenguaje de programación que se usará en este proyecto es VB .Net por lo que Visual Studio es la herramienta más adecuada.



- **MySQL** es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB — desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009 — desarrolla MySQL como software libre en un esquema de licenciamiento dual. (Oracle Corporation, 2014)

Por un lado, se ofrece bajo la licencia GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet.

El tipo de base de datos que se creará será una **base de datos relacional**, la cual consiste en un conjunto de tablas, a cada una de las cuales se le asigna un nombre exclusivo. Cada tabla tiene una estructura parecida a las entidades representadas en el modelo de clases.

Dado que cada tabla es un conjunto de dichas relaciones, hay una fuerte correspondencia entre el concepto de *tabla* y el concepto matemático de *relación*, del que toma su nombre el modelo de datos relacional. (Silberschatz, 2002)

## **Capítulo 1. Enfoque sistémico**

### **1.1 Datos generales de la institución**

#### **1.1.1 Misión**

La Universidad Nacional de Ingeniería es una Institución de la Educación Superior, estatal y autónoma, en búsqueda permanente de la excelencia académica, dedicada a formar profesionales en el campo de la Ciencia, la Ingeniería y la Arquitectura para que generen y difunden conocimientos con conciencia social, ética y humanística, con la finalidad de contribuir a la transformación tecnológica y al desarrollo sustentable de Nicaragua y la región Centroamericana.

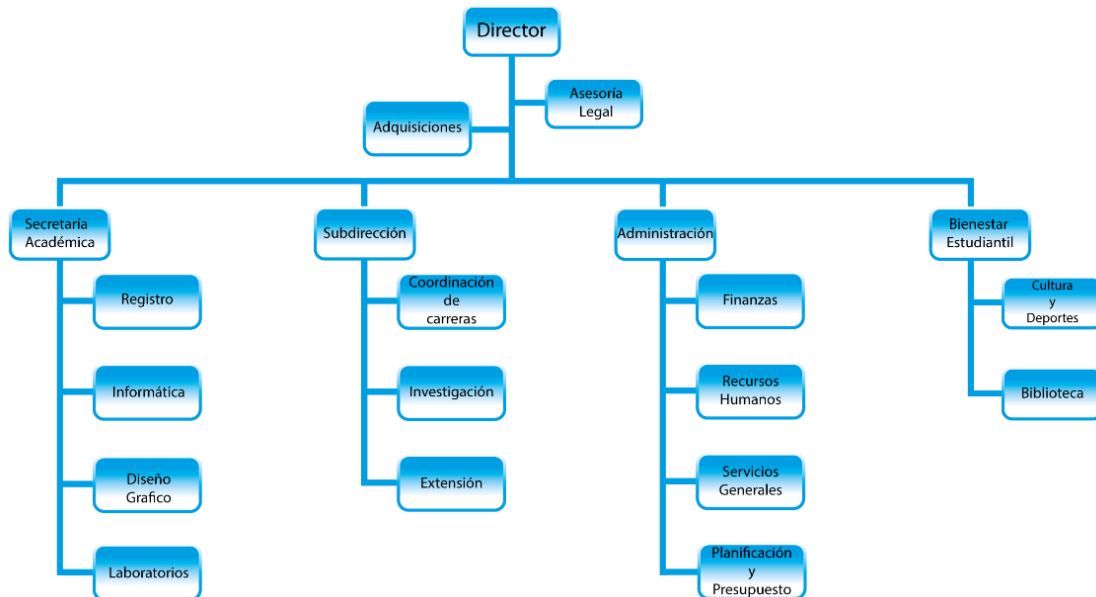
#### **1.1.2 Visión**

La Universidad Nacional de Ingeniería es una Institución que se consolida como líder nacional en la enseñanza de la Ingeniería y la Arquitectura, y es un referente en la investigación científica y tecnológica, construido mediante la interacción con los diversos actores y sectores sociales, económicos y culturales del país, contribuyendo al crecimiento y desarrollo nacional en función del bienestar de la sociedad nicaragüense.

#### **1.1.3 Objetivo**

Contribuir a la formación de profesionales en el ramo de la Ingeniería y la Arquitectura con alta calidad científica, tecnológica, humanística y compromiso social, facilitando el acceso y condiciones necesarias a los estudiantes de la región norte del país y además con el desarrollo de postgrados, formar especialistas altamente calificados.

#### 1.1.4 Estructura organizacional



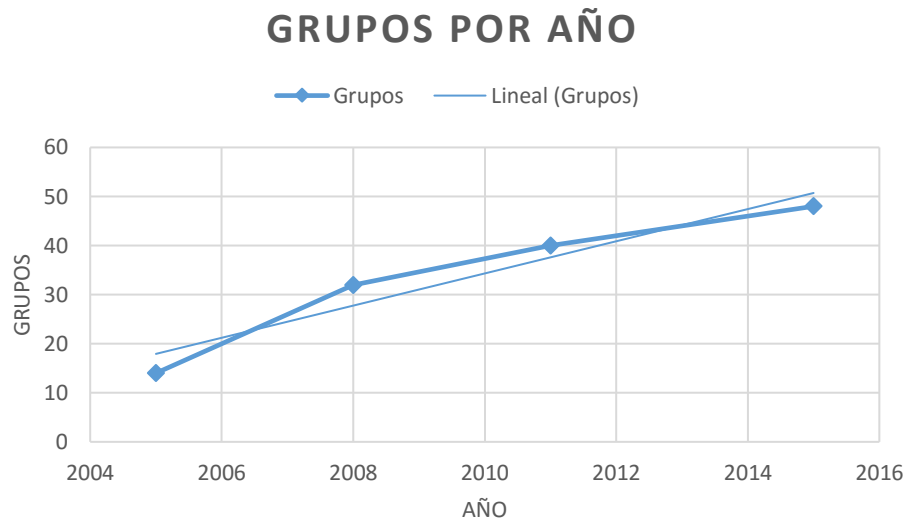
Estructura organizacional del Recinto Universitario Augusto C. Sandino.

Fuente: Sitio web oficial (<http://norte.uni.edu.ni/> )

### 1.2 Definición del problema

La complejidad del proceso de creación de horarios académicos está directamente relacionada a la cantidad de horarios a formar, a los recursos compartidos y a la cantidad de restricciones definidas.

En 2005 cuando inicia funciones el Recinto Universitario Augusto C. Sandino, ofrece a los estudiantes los turnos diurnos y sabatino, de acuerdo con la necesidad y características de la región, en total se crearon 14 grupos. En 2015 aún se ofrecen las mismas modalidades, sin embargo, el número de grupos asciende a 48, lo que representa un incremento del 242%.



Aunque en estos años se ha experimentado un incremento en la cantidad de grupos, la curva tiende a estabilizarse. Sin embargo, ya en este punto, la complejidad del proceso de creación de horarios es alta, 3 coordinadores deben formar horarios para 48 grupos que comparten recursos (aulas, medios audiovisuales y docentes), todo esto sin que se presenten conflictos por choque de horarios para grupos o docentes, teniendo en cuenta los pensum establecidos por carrera y considerando la duración del período y de cada asignatura.

Auxiliarse de herramientas que simplifiquen y agilicen esta tarea se convierte en una necesidad, ya que de lo contrario el tiempo y trabajo invertido crecería significativamente con cada iteración, hasta llegar a un punto en que la calidad del producto final se vería afectado por los errores cometidos.

Actualmente se usa un software para agilizar este proceso, sin embargo, es un programa de propósitos generales y no es adaptable a las necesidades específicas de la institución. Esto representa una pérdida de: tiempo, dinero, prestigio y trabajo, debido a que:

- Los horarios no se encuentran terminados en las fechas establecidas, presentan inconsistencias y deben ser modificados

- Los estudiantes y docentes no conocen a tiempo y con certeza sus horarios, por tanto, algunos deben viajar innecesariamente.
- La forma de trabajar del software utilizado actualmente no es igual a la forma de trabajar de la sede, lo que vuelve más complicado el proceso de planificación horaria, puesto que no permite realizar horarios por semestre, ni por turno, tampoco permite la reutilización de datos.
- Debido a que no trabaja en red, cada coordinador debe realizar su respectiva propuesta de manera individual, al llegar todas las propuestas a subdirección, es normal que ocurran choques horarios, siendo necesaria la revisión y corrección de propuestas.

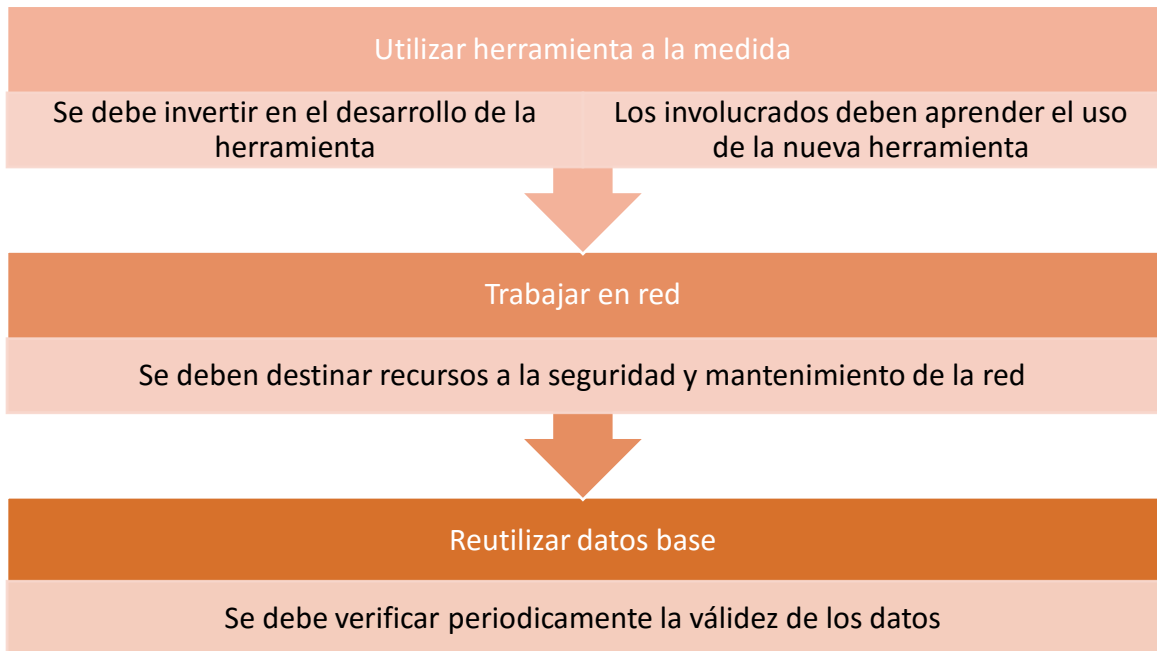
### 1.3 Árbol de problemas

Viajes innecesarios por estudiantes y docentes		Interrupción en las labores de los coordinadores
Tardanza en la entrega de horarios	Choque en horarios ya formados	Trabajo redundante
<b>Alto nivel de complejidad en la realización de horarios académicos</b>		
No se reutilizan datos	Se trabaja de forma aislada	No se usa una herramienta a la medida

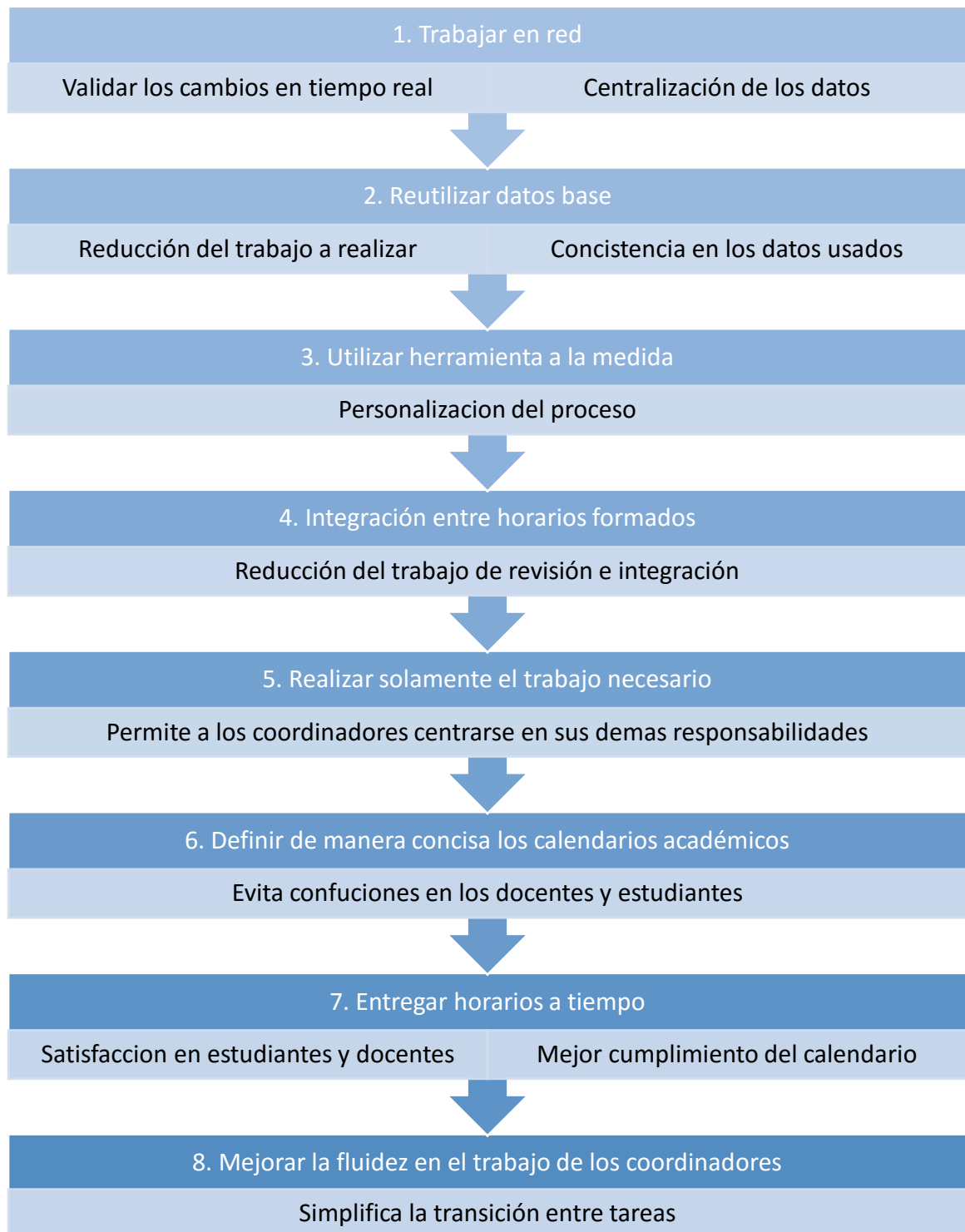
### 1.1 Árbol de objetivos

Definir de manera concisa los calendarios académicos		Mejorar la fluidez en el trabajo de los coordinadores
Entregar horarios a tiempo	Integración entre horarios formados	Realizar solamente el trabajo necesario
<b>Facilidad en la elaboración de horarios académicos</b>		
Reutilizar datos base	Trabajar en red	Utilizar herramienta a la medida

### 1.4 Consecuencias negativas



### 1.5 Consecuencias positivas



## 1.6 Grupos de interés

Quiénes son	Interés	Problema percibido	Demanda
<b>Estudiantes</b>	Conocer sus horarios de clase	<ul style="list-style-type: none"> <li>▪ Retraso al conocer horarios</li> <li>▪ Horarios cambian repentinamente</li> <li>▪ Viajes innecesarios</li> </ul>	<ul style="list-style-type: none"> <li>▪ Horarios entregados a tiempo</li> <li>▪ Consistencia de los horarios</li> </ul>
<b>Coordinación</b>	Elaborar horarios de manera eficiente	<ul style="list-style-type: none"> <li>▪ Pérdida de tiempo</li> <li>▪ Excesivo esfuerzo</li> </ul>	<ul style="list-style-type: none"> <li>• Tiempo y esfuerzo mínimos en la elaboración de horarios</li> </ul>
<b>Profesores</b>	Conocer sus horarios laborales	<ul style="list-style-type: none"> <li>▪ Retraso al conocer horarios</li> <li>▪ Horarios cambian repentinamente</li> <li>▪ Dificultad para adaptarse a nuevo horario</li> </ul>	<ul style="list-style-type: none"> <li>▪ Horarios entregados a tiempo</li> <li>• Consistencia de los horarios</li> </ul>
<b>Administración</b>	Cumplir con calendarios establecidos	<ul style="list-style-type: none"> <li>• Duplicación de trabajo</li> <li>• Desperdicio de recurso</li> </ul>	<ul style="list-style-type: none"> <li>• Eficiencia en la elaboración de horario</li> </ul>

## 1.7 Alternativas de solución

En correspondencia a los problemas tendientes a resolver, a los objetivos planteados y a las consecuencias positivas de los objetivos se han concebido las alternativas que apunten hacia la mejora de la situación descrita en el árbol del problema:

- 1) Adquirir un software de propósitos generales que sea consistente con la forma de trabajar de la sede.
- 2) Desarrollar un software propio ajustado a la realidad de la sede.
- 3) Desarrollar un método manual más eficiente para la gestión horaria.



- 4) Separar los recursos por coordinación/carrera.

### 1.8 Evaluación de soluciones

Se evalúa el cumplimiento de objetivos para cada alternativa planteada, definiendo el nivel de viabilidad y costo para obtener un resultado cuantitativo que refleje la relación beneficio-costos.

	O1	O2	O3	O4	O5	O6	O7	O8	Total	Viabilidad	Costo	Resultado
A1	1	1	0	1	0	1	1	1	6	3	3	6
A2	1	1	1	1	1	1	1	1	8	3	2	12
A3	0	0	1	1	0	1	1	0	4	2	1	8
A4	0	0	0	0	1	1	1	1	4	1	3	1.33

- **Total:** cantidad de objetivos cubiertos por la alternativa de solución. Mínimo = 0, Máximo = 8.
- **Viabilidad:** impedimentos a la implementación de la alternativa de solución, a mayor impedimento, menor viabilidad. Mínimo = 1, Máximo = 3.
- **Costo:** cantidad de recursos que deben invertirse para implementar la alternativa de solución, a mayor cantidad de recursos, mayor costo. Mínimo = 1, Máximo = 3.
- **Resultado:** valor cuantitativo que refleja la relación beneficio-costos.  

$$\text{Resultado} = \text{Total} * \text{Viabilidad} / \text{Costo}.$$

La alternativa de solución mejor puntuada ha sido la A2 (Desarrollar un software propio ajustado a la realidad de la sede), esto se debe a que aporta el mayor beneficio (8 de 8) a un costo razonable, además no presenta impedimentos en su implementación (ver detalle de la factibilidad en el Capítulo 2).

## **Capítulo 2. Estudios de factibilidad**

El desarrollo e implementación de un Sistema para la Planificación de Horarios Académicos en la Universidad Nacional de Ingeniería, Recinto Universitario Augusto C. Sandino resulta factible desde el punto de vista económico, técnico y operacional puesto que la sede presenta todas las condiciones para la realización del proyecto.

### **2.1 Factibilidad técnica**

#### **2.1.1 Red local**

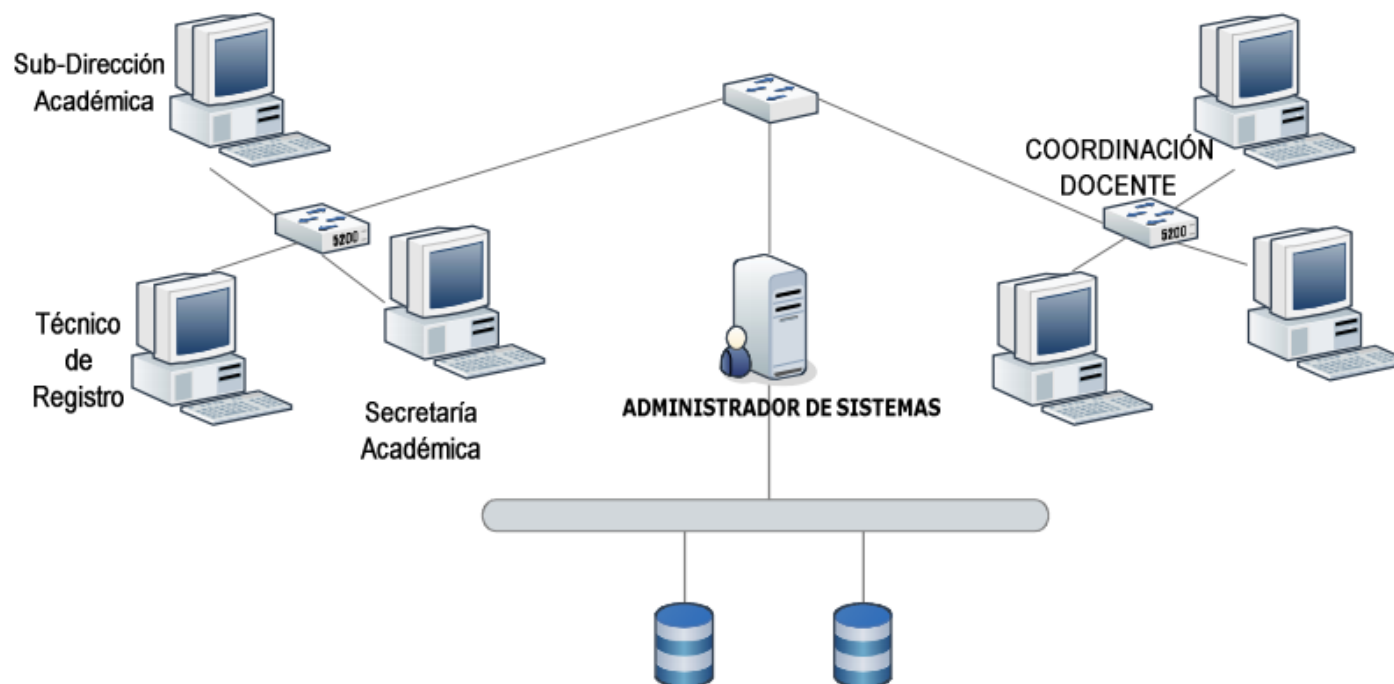
Dentro de los requerimientos para la implementación del sistema propuesto, es necesario que los equipos de cómputo del área de subdirección y coordinación se encuentren conectados mediante una intranet para gestionar la información.

Actualmente este requerimiento está satisfecho en la sede, ya que se cuenta con una red de computadoras conectadas en todo el edificio, esta red conecta las diferentes áreas de la sede incluidas subdirección y coordinación.

La red disponible se encuentra segmentada en *vlangs* para separar y controlar los accesos a la red, de modo que la red destinada a estudiantes y las redes libres se encuentran completamente separada de las demás, lo cual es necesario para fortalecer la seguridad de los sistemas informáticos usados en la institución.

Los nodos de subdirección y coordinación se encuentran conectados directamente (a través de *vlangs*) por lo que no es necesario realizar modificaciones o nuevas implementaciones en la red local.

## Sistema para Planificación de Horarios Académicos



Segmento de red local del Recinto Universitario Augusto C. Sandino.

Fuente: Área de informática del recinto.

### 2.1.2 Equipos de cómputo

Se requieren dos tipos de equipos de cómputo para la implementación del sistema propuesto, uno que funcione como cliente (los equipos de los coordinadores de carrera) y otro que funcione como servidor (para alojar la base de datos). Se muestra a continuación los requerimientos mínimos, recomendados y existentes para cada uno ellos.

#### Equipos clientes requeridos

Mínimo	Recomendado
<b>Hardware</b>	
HDD: 500 MB libre	HDD: 2 GB Libre
RAM: 1 GB	RAM: 4 GB
CPU: Dual Core 2.2 GHz	CPU: I3-6100
Teclado	
Mouse	
Monitor	
Impresora (local o en red)	
Conexión de red	
<b>Software</b>	
Net Framework 4.0	
Windows XP	Windows 10
	Lector de pdf
	Suite de ofimática para leer archivos: .docx, .xlsx, .csv
	Navegador web
	Cliente de correo

### Equipo servidor requerido

Mínimo	Recomendado
<b>Hardware</b>	
HDD: 10 GB libre	HDD: 50 GB Libre
RAM: 2 GB	RAM: 4 GB
CPU: Dual Core 2.2 GHz	CPU: I5-6500
Conexión de red	
<b>Software</b>	
MySQL 5.5 o superior	
Windows Server/Linux	

### Equipos disponibles

Cliente	Servidor
<b>Hardware</b>	
HDD: 1.5 TB libre	HDD: 1 TB Libre
RAM: 16 GB	RAM: 8 GB
CPU: I5-6500	CPU: Xeon E7
Teclado	
Mouse	
Monitor	
Impresora en red	
Conexión de red	
<b>Software</b>	
Net Framework 4.0	Debian 8.0
Windows 10	MySQL 5.7
Adobe Reader	

## Sistema para Planificación de Horarios Académicos

MS Office 2016	
Google Chrome	

El Recinto Universitario Augusto C. Sandino cuenta con todos los equipos necesarios para ejecutar el sistema con fluidez, dispone de un servidor completamente funcional, un área de docentes con 4 computadoras conectadas en red, un área de coordinación con 5 computadoras y 1 impresora en red.

Se cuenta además con un área destinada al mantenimiento, actualización y reparación de los equipos informáticos en general (PC, redes, equipos de conexión, impresoras, etc.), esta se encarga de realizar mantenimientos preventivos y correctivos, actualizar el software utilizado y solicitar la compra de repuestos al área correspondiente.

### **2.2 Factibilidad económica**

Desde el punto de vista material se requieren en total 4 computadoras (3 PC en el área de coordinación, 1 PC en el área de subdirección), un servidor donde se alojará la base de datos, una impresora y conexión de red que comunique a los distintos equipos.

El personal encargado de manipular el sistema serán los coordinadores de carreras en conjunto con el área de subdirección. Los reportes extraídos podrán ser enviados y usados por otras áreas de la institución, sin embargo, estas últimas no necesitan acceso directo al mismo. La administración del servidor y la base de datos estará a cargo del área de informática.

El Recinto Universitario Augusto C. Sandino actualmente cuenta con todos estos recursos y el personal necesario para la gestión del sistema, por lo cual no necesita invertir en nuevos equipos ni en contratar personal adicional.

El desarrollo del sistema propuesto no genera gastos de licencias para la institución ya que no es necesario adquirir software de pago, y las herramientas necesarias para el funcionamiento del sistema se encuentran actualmente disponibles en la sede.

### 2.2.1 Estimación de costo del sistema

El costo estimado del sistema asciende a \$9,677.74 (nueve mil seiscientos setenta y siete dólares con setenta y cuatro centavos) según el modelo constructivo de costos (COCOMO).

#### Características del dominio de información (PF)

	Simple	Media	Alta	Total
Números de entrada de usuario	38 * 3	6 * 4	10 * 6	198
Número de salidas de usuario	12 * 3	2 * 4	1 * 6	50
Número de peticiones de usuario	0 * 3	0 * 4	0 * 6	0
Número de archivos lógicos	9 * 3	1 * 4	1 * 6	37
Número de interfaces externas	10 * 3	5 * 4	0 * 6	50
Total de puntos de función				335
Multiplicador				1.15
<b>Total PF</b>				<b>385</b>

#### Escala de multiplicación (Fi)

Condición	Valor
¿Requiere el software copias de seguridad y recuperación fiable?	1
¿Se requiere comunicación de datos?	4
¿Existen funciones de procesamiento distribuido?	4

## Sistema para Planificación de Horarios Académicos

¿Es crítico el rendimiento?	2
¿se ejecutará el sistema en un entorno operativo existente y fuertemente utilizado?	5
¿Requiere entradas de datos interactivas?	2
¿Requiere la entrada de dato interactiva que las transacciones de entrada se lleven a cabo en múltiples pantallas u operaciones?	0
¿Se actualizará el archivo maestro de forma interactiva?	3
¿Son complejas las entradas, salidas, archivos o peticiones?	4
¿Es complejo el procesamiento interno?	4
¿Se ha diseñado el código para que sea reutilizable?	3
¿Están incluidos en el diseño la conversión e instalación?	3
¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	3
¿Se ha diseñado el sistema para facilitar los cambios y para ser fácilmente usado por el usuario?	3
<b>Total Fi</b>	<b>41</b>

### Escala

0 - No influye

1 - Incidente

2 - Moderado

3 - Media

4 - Significativa

5 - Esencial



## Factores de Escala (SFi)

PREC: Desarrollos previos similares

- 0.00: nuevo desarrollo es idéntico a previos
- 1.24: es muy parecido
- 2.48: bastante parecido
- **3.72: aspectos novedosos**
- 4.96: muy diferente
- 6.20: totalmente diferente.

FLEX: Flexibilidad del desarrollo

- 0.00: metas son generales = 2.03
- 1.01: cierto acuerdo
- **2.03: acuerdo general**
- 3.04: cierta flexibilidad
- 4.05: flexibilidad ocasional
- 5.07: riguroso

RESL: Manejo de riesgos y arquitectura

- 0.00: plan identifica todos los riesgos críticos y establece hitos para resolverlos, calendario y presupuesto toma en cuenta riesgos, arquitectura puede tomarse hasta el 40% del esfuerzo de desarrollo, herramientas disponibles para resolver/mitigar riesgos y verificar especificación de la arquitectura muy poca incertidumbre de remisión, interfaz con usuario, tecnología, desempeño, riesgos no son críticos.
- 1.41: plan identifica la mayoría de los riesgos críticos y establece hitos para resolverlos, calendario y presupuesto toma en cuenta la mayoría de los riesgos, arquitectura puede tomarse hasta el 33% del esfuerzo de desarrollo, herramientas disponibles para resolver/mitigar mayoría de riesgos y verificar especificación de la arquitectura, poca incertidumbre remisión, interfaz con usuario, tecnología, desempeño, riesgos no son críticos.
- **2.83: plan identifica muchos de los riesgos críticos y establece hitos para resolverlos, calendario y presupuesto generalmente toma en cuenta riesgos, arquitectura puede tomarse hasta el 25% del esfuerzo**

**de desarrollo, herramientas regularmente disponibles para resolver/mitigar riesgos y verificar especificación de la arquitectura algo de incertidumbre remisión, interfaz con usuario, tecnología, desempeño, no más de un riesgo crítico.**

- 4.24: plan identifica algunos de los riesgos críticos y establece hitos para resolverlos, calendario y presupuesto toma en cuenta algunos de los riesgos, arquitectura puede tomarse hasta el 17% del esfuerzo de desarrollo, hay problemas con la disponibilidad del arquitecto, algo de herramientas disponibles para resolver/mitigar riesgos, verificar especificación de la arquitectura, considerable incertidumbre re misión, interfaz con usuario, tecnología, desempeño, entre 2-4 riesgos críticos.

5.65: plan identifica pocos riesgos críticos y establece hitos para resolverlos, calendario y presupuesto toma en cuenta pocos riesgos, arquitectura puede tomarse hasta el 10% del esfuerzo de desarrollo, hay problemas con la disponibilidad del arquitecto (disp. menor al 40%), pocas herramientas disponibles para resolver/mitigar riesgos y verificar especificación de la arquitectura, significativa incertidumbre re misión, interfaz con usuario, tecnología, desempeño, entre 5-10 riesgos críticos.

- 7.07: plan no identifica los riesgos críticos, calendario y presupuesto no toma en cuenta los riesgos, arquitectura puede tomarse hasta el 5% del esfuerzo de desarrollo, hay problemas con la disponibilidad del arquitecto (disp. menor del 20%), herramientas no disponibles para resolver/mitigar riesgos y verificar especificación de la arquitectura, extrema incertidumbre remisión, interfaz con usuario, tecnología, desempeño, más de 10 riesgos críticos.

TEAM: Cohesión del equipo de desarrollo

- **0.0: interacciones fluidas, objetivos y culturas de accionistas totalmente consistentes, total habilidad y disponibilidad de accionistas para acomodar objetivos de otros accionistas, dilatada experiencia previa operando como equipo, visión y compromisos 100% compartidos.**

- 1.1: interacciones altamente cooperativas, objetivos y culturas de accionistas fuertemente consistentes, fuerte habilidad y disponibilidad de accionistas para acomodar objetivos de otros accionistas, considerable experiencia previa operando como equipo, visión y compromisos considerablemente compartidos.
- 2.19: interacciones principalmente cooperativas, objetivos y culturas de accionistas considerablemente consistentes, considerable habilidad y disponibilidad de accionistas para acomodar objetivos de otros accionistas, mediana experiencia previa operando como equipo, visión y compromisos medianamente compartidos.
- 3.29: interacciones básicas cooperativas, objetivos y culturas de accionistas básicamente consistentes, habilidad y disponibilidad básica de accionistas para acomodar objetivos de otros accionistas, poca experiencia previa operando como equipo, visión y compromisos poco compartidos.
- 4.38: algunas interacciones difíciles, objetivos y culturas de accionistas algo consistentes, algo habilidad y disponibilidad de accionistas para acomodar objetivos de otros accionistas, poca experiencia previa operando como equipo, visión y compromisos poco compartidos.
- 5.48: interacciones difíciles, objetivos y culturas de accionistas poco consistentes, poca habilidad y disponibilidad de accionistas para acomodar objetivos de otros accionistas, nada de experiencia previa operando como equipo, visión y compromisos nada compartidos.

CMM: Modelo de Capacidad de Madurez

Nivel	Característica	Desafíos claves
Optimizante 0	Mejoramiento realimentado al proceso	Un proceso humano-intensivo Mantiene la organización en nivel optimizante
<b>Administrado 1.56</b>	(Cualitativo) Proceso medido	Cambio de tecnología Análisis de procesos Prevención de problemas
Definido 3.12	(Cualitativo) Proceso definido e institucionalizado	Métricas de procesos Análisis de procesos Planes cuantitativos de calidad
Repetible 4.68	(Intuitivo) Proceso dependiente de individuos	Entrenamiento, testeo Prácticas técnicas y revisiones Foco en el proceso, estándares y procesos
Inicial 7.8	(Ad hoc/caótico)	Administración de proyectos y planificación Administración de la configuración Aseguramiento de la calidad de software

**Total SFI = 10.14**

### Factor de esfuerzo compuesto (EMi)

RELY (Seguridad Requerida)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador RELY</b>	Efecto de falla sin ninguna consecuencia.	Efecto Peq. Recuperable fácilmente.	Fallas Moderadas.	Grandes Pérdidas Financieras	Riesgo de Vidas Humanas
<b>Valor Asociado</b>	0.75	<b>0.88</b>	1.00	1.15	1.39

DATA (Tamaño de Base de Datos)

	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador DATA</b>	<10	>=10 Y <100	>=100 Y <1000	>=1000
<b>Valor Asociado</b>	0.93	<b>1.00</b>	1.09	1.19

DOCU (Documentación Adaptada al Ciclo de Vida)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador DOCU</b>	Muchas Etapas sin cobertura.	Algunas Etapas sin Cobertura.	Adaptado a las etapas del Ciclo de Vida.	Excesiva Documentación.	Muy Excesiva Documentacion.
<b>Valor Asociado</b>	0.89	0.95	<b>1.00</b>	1.06	1.13

CPLX (Complejidad)

NIVEL	OPERACIONES DE CONTROL	OPERACIONES MATEMÁTICAS	OPERACIONES DE ENTRADA/SALIDA	OPERACIONES DE MANEJO DE DATOS	Valor Asociado
<b>MUY BAJO</b>	Códigos lineales: DO IF-THEN-ELSE Predicados simples, pocas subrutinas.	Evaluación de expresiones matemáticas simples: $C = A+B*(D-E)$ .	Lecturas simples Escrituras con formatos simples.	Arreglos simples en memoria RAM.	0.75
<b>BAJO</b>	Subrutinas en secuencia la mayor parte en predicados simples.	Evaluación de expresiones reiteradas. Raíces y Potencias.	No se necesitan procesos especiales de E/S. Sólo toma y entrega de información. No hay solapamiento.	Archivos simples sin cambios en la estructura de datos.	0.88
<b>NOMINAL</b>	Programación Estructurada (PE). Mayormente subrutinas simples. Tablas de decisión.	Uso de subrutinas matemáticas y estadísticas. Operaciones con matrices y vectores.	E/S comprende selecciones, chequeos de estado y tratamiento de errores.	Múltiples archivos de E/S. Cambios simples en la estructura de datos.	<b>1.00</b>
<b>ALTO</b>	Programa estructurado con muchas subrutinas. Considerables módulos. Colas. Pilas.	Análisis numérico. Interpolación multivariable. Ecuaciones diferenciales.	Optimización del solapamiento de E/S. Operaciones de E/S a nivel físico.	Complejas reestructuraciones de los datos. Subrutinas activadas por el FD.	1.15
<b>MUY ALTO</b>	Código reentrante y recursivo. Prioridad fija de interrupción manual.	Ecuaciones con matrices singulares. Ecuaciones diferenciales parciales. Análisis numérico difícil.	Subrutinas para interrumpir el servicio. Manejo de líneas de comunicación.	Uso generalizado de lo anterior. Archivo comando de procesamiento. Optimización de búsqueda.	1.30
<b>EXTRA ALTO</b>	Programación múltiple. Cambios dinámicos de prioridad. Micro código.	Análisis numérico difícil y no estructurado. Análisis muy preciso. Métodos estocásticos.	Operaciones micro programables.	Dirección de datos en lenguaje natural. Estructuras dinámicas altamente enlazadas	1.66

RUSE (Reutilización Requerida)

	BAJO	NOMINAL	ALTO	MUY ALTO	EXT.ALTO
<b>Indicador RUSE</b>	Ninguna	A través del Proyecto	A través de Programas	A través de Líneas de Productos.	A través de Líneas Múltiples de Prod.
<b>Valor Asociado</b>	0.91	<b>1.00</b>	1.14	1.29	1.49

TIME (Tiempo de Ejecución Requerido)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO	EXT. ALTO
<b>Indicador TIME</b>			50%	70%	85%	95%
<b>Valor Asociado</b>	1.00	1.00	<b>1.00</b>	1.11	1.31	1.67

STOR (Almacenamiento principal Requerido)

	BAJO	NOMINAL	ALTO	MUY ALTO	EXT. ALTO
<b>Indicador STOR</b>		50%	70%	85%	95%
<b>Valor Asociado</b>	1.00	<b>1.00</b>	1.06	1.21	1.57

PVOL (Volatilidad de la Plataforma)

	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador PVOL</b>	>=1 MES Y <=12 MESES	>=6 MESES Y <=2 SEM	>=2 MESES Y <=1 SEM	>=2 SEM Y <= 2 DIAS
<b>Valor Asociado</b>	<b>0.87</b>	1.00	1.15	1.30

ACAP (Capacidad del Analista)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador ACAP</b>	15 %	35%	55%.	75%	90%
<b>Valor Asociado</b>	1.50	1.22	1.00	<b>0.83</b>	0.67

AEXP (Experiencia del Analista)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador AEXP</b>	2 meses	6 meses	12 meses	36 meses	72 meses
<b>Valor Asociado</b>	1.22	1.10	<b>1.00</b>	0.89	0.81



PCAP (Capacidad del programador)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador PCAP</b>	15 %	35%	55%.	75%	90%
<b>Valor Asociado</b>	1.37	1.16	1.00	<b>0.87</b>	0.74

PEXP (Experiencia en la Plataforma de Sistema Operativo)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador PEXP</b>	2 meses	6 meses	12 meses	36 meses	72 meses
<b>Valor Asociado</b>	1.25	1.12	1.00	0.88	<b>0.81</b>

LTEX (Experiencia en Lenguaje y Herramienta)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador PEXP</b>	2 meses	6 meses	12 meses	36 meses	72 meses
<b>Valor Asociado</b>	1.22	1.10	1.00	<b>0.91</b>	0.84

PCON (Continuidad del personal)

	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador PCON</b>	48%	24%	12%	6%	3%
<b>Valor Asociado</b>	1.24	1.10	1.00	0.92	<b>0.84</b>

TOOL (Uso de Herramientas de SW)

INDICADOR	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO
<b>Indicador TOOL</b>	Editar, Codificar y Corregir.	Ciclos y Pequeña Integración.	Integración Moderna.	Bastante  Integración.	Cuantiosa Integración.
<b>Valor Asociado</b>	1.24	1.12	<b>1.00</b>	0.86	0.72

SITE (Desarrollo Multitarea)

INDICADOR	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO	EXT. ALTO
<b>Indicador SITE</b>	Teléfono, Correo.	Teléfono , Fax.	Banda Corta, Emails.	Banda Ancha	Banda Ancha, Ocasionalmente Video Conferencia.	Múltiples formas, Interactivo.
<b>Valor Asociado</b>	1.25	1.10	1.00	0.92	<b>0.84</b>	0.78

SCED (Esquema de Desarrollo Programado)

INDICADOR	MUY BAJO	BAJO	NOMINAL	ALTO	MUY ALTO	EXT. ALTO
<b>Indicador SCED</b>	<b>75% del Nominal.</b>	<b>85%</b>	<b>100%</b>	<b>130%</b>	<b>160%</b>	
<b>Valor Asociado</b>	1.29	1.10	<b>1.00</b>	1.00	1.00	1.00

**Total EMI = 0.2875**

Resumen de cálculos	
<b>PF</b>	385
<b>Fi</b>	41
<b>SFI</b>	10.14
<b>EMI</b>	0.2875

$$\text{PFA} = \text{PF} \cdot (0.65 + 0.01 \sum \text{Fi})$$

$$\text{PFA} = 385 (0.65 + 0.01(41)) = 408.1$$

$$\text{TLDC} = 15 \cdot \text{PFA}$$

$$\text{TLDC} = 15 \cdot 408.1 = 6,121$$

$$\text{B} = 0.91 + 0.01 \times \sum \text{SFi},$$

$$\text{B} = 0.91 + 0.01 \times 10.14 = 1.0114$$

$$\text{Esfuerzo (personas)} = 2.94 \times (\text{TLDC})^{\text{B}} \times \Pi \text{ EMI}$$

$$\text{Esfuerzo} = 2.94 \times (6,121)^{1.0114} \times 0.2875 = 5$$

$$\text{Tiempo (meses)} = 3.67 \cdot \text{Esfuerzo}^{0.28 + 0.002 \cdot \sum \text{SFi}}$$

$$\text{Tiempo} = 3.67 \cdot 5^{0.28 + 0.002 \cdot 10.14} = 6$$

Según los cálculos obtenidos del modelo constructivo de costos, se estima una duración de desarrollo de seis meses, con cinco personas trabajando en el proyecto. Se detalla a continuación los costos estimados para ejecutar el proyecto siguiendo los parámetros calculados.

## Salarios

Cargo	Salario (mes)	Salario (día)
Analista	\$ 400	13.33
Programador * 3	\$ 250	8.33
Tester	\$ 200	6.66

## Etapas

	Recopilación de Información	Análisis	Diseño y desarrollo	Documentación	Pruebas	Total, Meses
Analista	1	1	3	0.5	0.5	6
Programador			3	0.5	0.5	4
Tester				0.5	0.5	1

## Gasto total

	Salario total	Prestaciones	INSS Patronal	INATEC	Total
Analista	2,400.00	600.00	444.00	48.00	3,492.00
Programador	1,000.00	250.00	185.00	20.00	1,455.00
Tester	200.00	50.00	37.00	4.00	291.00

$$\text{CFT} = 3,492.00 + (1,455.00 * 3) + 291 = 8,148.00$$

## CMat:

Artículo	Precio
1 Memoria USB 8 Gb	C\$ 300
1 Caja de lapiceros	C\$ 60
1 Caja de lápices de grafito	C\$ 30
4 Libretas	C\$ 100

1 Resma de papel blanco	C\$ 100
1 Caja de grapas	C\$ 30
1 Caja de correctores	C\$ 75
1 Caja de resaltadores	C\$ 55
1 Caja de borradores	C\$ 15
1 Caja de clip	C\$ 12
<b>TOTAL</b>	<b>C\$ 777   \$ 26</b>

### Otros Gastos

- Transporte y alimentación= **\$417.54**

$$\text{CUMT} = (\Sigma \text{Ck}/\text{Prk}) * \text{HTMk} + \Sigma(\text{Cek} * \text{HTMk} * \text{CKHk}) + \Sigma \text{CRMk}$$

**Ck:** Costo del equipo = \$ 1,950

**Prk:** Periodo de recuperación horas =  $(4 * 365 * 24) = 35,040$  horas

**Cek:** Consumo de energía = 0.1794 kw/hora

**HTMk:** Horas tiempo maquina = tiempo de desarrollo \* cantidad días trabajados al mes \* cantidad horas trabajadas diarias =  $6 * 20 * 8 = 960$  horas

**CKHk:** Costo kw/hora = \$0.1219 kw/h

**CRMk:** Costo del mantenimiento preventivo = \$30

$$\text{CUMT} = (1,950/35,040) * 960 + (0.1794 * 960 * 0.1219) * 3 + 30 * 3 = \$206.41$$

$$\text{CD} = \text{CFT} + \text{CUMT} + \text{CMAT} + \text{OG}$$

$$\text{CD} = \$8,148.00 + 206.41 + 26 + 417.54 = \$8,797.95$$

$$\text{CI} = 10\% \text{ CD}$$

$$\text{CI} = 0.1 * 8,797.95$$

$$\text{CI} = 879.79$$

$$\text{CT} = \text{CD} + \text{CI}$$

$$\text{CT} = \$8,797.95 + \$879.79$$

$$\text{CT} = \mathbf{\$9,677.74}$$

## **2.3 Factibilidad operativa**

### **2.3.1 Capital humano**

Es necesario disponer de personal capacitado para instalar, operar y mantener el sistema propuesto, se requiere que el personal operativo (coordinadores y subdirección) se encuentren familiarizados con el uso de herramientas informáticas para la solución de problemas, deben manejar paquetes ofimáticos y tener conocimientos básicos del sistema operativo Windows. El personal encargado de la instalación y mantenimiento del software, deberá contar con conocimientos en el área de reparación y mantenimiento de PC, consultas SQL y administración de servidores Linux.

El Recinto Universitario Augusto C. Sandino actualmente cuenta con personal altamente capacitado para las tareas descritas anteriormente, todos los involucrados poseen al menos estudios de grado superior, además son capacitados cada año en el uso de herramientas informáticas para agilizar sus responsabilidades. Se encuentran familiarizados en el uso de software similar al propuesto, tales como, el Sistema de Registro Académico (SIRA) y el Portal de Administración de Laboratorios en Línea. El personal encargado de la administración del servidor y base de datos, cuenta además con varios años de experiencia en la administración de servidores Linux y base de datos relacionales.

Los principales usuarios de la aplicación serán los coordinadores de carrera, ya que ellos serán quienes ingresen la mayoría de los datos, además de crear los horarios y generar los reportes necesarios para entregar información a los demás interesados. Se muestran a continuación elementos de la ficha ocupacional del puesto de coordinador de carrera que deben ser tomados en cuenta durante la adopción del sistema propuesto.

### **Ficha ocupacional**

**Título del puesto:** Coordinador.

**Departamento:** -

**Jefe superior inmediato:** Sub-Director

#### **I. Funciones.**

- a. Realizar horarios académicos
- b. Gestionar actividades relacionadas con la carrera que coordina.

#### **II. Requisitos del puesto.**

##### **1. Habilidad.**

###### **a. Educación:**

Haber realizado estudios universitarios.

###### **b. Experiencia:**

Tener 3 años de experiencia como docente.

###### **c. Iniciativa e ingenio.**

Debe tener capacidad de resolver de manera eficaz y eficiente los problemas que se le presentan en el puesto de trabajo.

##### **2. Esfuerzo.**

###### **a. Mental:**

Se esfuerza mentalmente en la elaboración de los distintos esquemas para la planeación, se requiere de una atención intensa.

Cansancio mental fuerte por los largos periodos de análisis y diseño de los proyectos a desarrollar.

###### **b. Físico:**

Ninguno.

### **3. Responsabilidad**

#### **a. Equipo.**

Tiene responsabilidad sobre el equipo de cómputo y de oficina que le es asignado (como los muebles y otros) durante el periodo de tiempo que lo esté utilizando.

#### **b. Materiales o productos.**

Ninguno.

#### **c. Trabajo de otros.**

Supervisa el trabajo de las personas a cargo (Docentes).

### **4. Condiciones de trabajo.**

#### **a. Ambiente de trabajo:**

Ambiente con excelente ventilación, iluminación, colores que relajan la vista y sin ruido excesivo.

#### **b. Riesgo:**

Ambiente de trabajo normal, la posibilidad que ocurra un accidente es eventual.

#### **2.3.2 Complejidad de adopción**

Con el fin de mejorar la experiencia de usuario, la interfaz gráfica y los mecanismos de control del sistema propuesto se asemejan a los usados en aplicaciones existentes. Esto reducirá significativamente la curva de aprendizaje de los involucrados y ayudará a que la integración del sistema a las tareas diarias sea más fácil y fluida, reduciendo la oposición al cambio.

Debido al alto nivel de habilidad y conocimientos de los involucrados y a la baja complejidad en el manejo del sistema, no se prevé que surjan mayores dificultades durante la adopción del mismo, sin embargo, previo al periodo de uso oficial, se realizarán capacitaciones con todos los involucrados, en las que se les mostrará



como operar el sistema y como resolver los conflictos que se puedan presentar en cada escenario.

El Recinto Universitario Augusto C. Sandino cuenta actualmente con manuales de funciones bien definidos, en los que se detallan los responsables y los procedimientos involucrados en la creación de horarios académicos. Al implementar el sistema propuesto, no se afectan ninguna de las funciones, ya que simplemente se está agregando una herramienta informática para realizar una tarea ya establecida en las funciones del personal. Por lo que el sistema propuesto, no desplaza al personal existente, ni representa un cambio repentino a la realización de sus labores.

## Capítulo 3. Análisis de requerimientos

### 3.1 Descripción del proceso

La planificación de horarios académicos es el proceso mediante el cual se le asigna a cada grupo de clases un turno, aula y un calendario semanal que especifica las asignaturas a cursar, el docente que las impartirá, las repeticiones y las horas de inicio y fin.

Los coordinadores de carrera, crean una propuesta de horarios que es enviada a subdirección, donde es autorizada por el subdirector. Los individuos involucrados en este proceso, son:

- **Coordinadores:** Son los encargados de elaborar la planificación horaria, también ingresan los datos extras requeridos en el proceso, los cuales incluyen: docentes, aulas, grupos, carreras, turnos y horas.
- **Subdirector:** Es el encargado de aprobar o rechazar la propuesta horaria elaborada por los coordinadores.

El análisis de requerimientos brindará la información necesaria para detallar las necesidades de los usuarios. Después de concluir esta etapa, todos los involucrados en el desarrollo podrán comprender las necesidades y requerimientos a cubrir. Para la recopilación de la información necesaria en el análisis de requerimientos se llevaron a cabo reuniones con las partes interesadas.

### 3.2 Administración del sistema

#### 3.2.1 Definición de parámetros

La definición de parámetros del sistema es realizada por los coordinadores de carrera. Esta función consiste en definir la información correspondiente a Aulas,

Medios audiovisuales, Facultades, Carreras, Planes de estudio, Grupos, Turnos y Horas.

Dado que el sistema propuesto almacena esta información de manera centralizada en una base de datos accesible para todos los interesados (a través del sistema propuesto), los coordinadores deberán ingresar la información una sola vez y posteriormente actualizar los aspectos que varíen.

### **3.2.2 Creación de cuentas y permisos**

El sistema propuesto será de uso restringido, únicamente tendrán acceso a este, los usuarios que estén debidamente registrados y habilitados para ello. La creación de usuarios y permisos la realizará el administrador del sistema, el cual será designado por el sub director.

Las cuentas de usuarios deberán cumplir con los siguientes criterios:

- Se creará una cuenta de usuario por cada persona que tendrá acceso al sistema. Bajo ninguna circunstancia varias personas compartirán una cuenta, o tendrán más de una cuenta asociada.
- Los permisos se definirán a través de roles, no directamente sobre cada usuario.
- Cada cuenta deberá tener asignado al menos un rol.
- Se deberá diferenciar entre administradores y usuarios.
- Los administradores tendrán acceso completo al sistema.
- Las claves de acceso deberán almacenarse encriptadas.
- Las cuentas pueden ser deshabilitadas y habilitadas por el administrador, sin que esto afecte los horarios ya almacenados.

### **3.3 Formación de horarios**

Los horarios académicos son elaborados a partir de los parámetros previamente definidos en el sistema, estos datos son combinados a conveniencia por los

coordinadores de carrera, sin embargo, deben cumplir una serie de criterios mínimos de aceptación:

- Cada coordinador tendrá asignada al menos una carrera.
- Los horarios de una determinada carrera solo podrán ser creados y editados por el coordinador asignado a la carrera.
- Un grupo de clases no puede tener asignada más de un aula simultáneamente.
- Un docente no puede impartir más de una asignatura a un grupo simultáneamente.
- Cada grupo tiene un turno oficial asignado.
- Un docente no puede tener asignado más de un grupo a la misma hora.
- Un grupo no puede tener asignada más de una asignatura a la misma hora.
- Un aula no puede tener asignado más de un grupo a la misma hora.
- Los horarios se renuevan por periodos (semestral para el turno diario y trimestral para el turno sabatino)
- Cada horario deberá cumplir con las horas clase predeterminadas en el pensum (el cumplimiento es aproximado, de modo que no siempre se alcanza el 100% de exactitud en las horas asignadas).

### **3.4 Generación de informes**

El sistema propuesto deberá generar informes sobre los diferentes datos almacenados, principalmente los relacionados a horarios académicos. La finalidad de esta función es entregar información oportuna y precisa a los interesados. Los principales informes son:

- Horario completo de un docente específico.
- Horario completo de un grupo específico.
- Uso de aulas durante la semana.
- Resumen de los parámetros del sistema (carreras, grupos, turnos, facultades, aulas, asignaturas, docentes.)

### 3.5 Resumen de requerimientos

Nº	Requerimiento	Descripción	Criterio de aceptación
R1	Registro de facultades	Almacenar, recuperar y editar los datos generales del módulo.	Se almacena la información digitada. Se permite el filtro y búsqueda en datos existentes.
R2	Registro de aulas		
R3	Registro de asignaturas		
R4	Registro de docentes		
R5	Registro de grupos		
R6	Registro de carreras		
R7	Registro de períodos		
R8	Registro de turnos		
R9	Registro de pensum		
R10	Formar horarios	Almacenar, recuperar y editar horarios académicos usando la información previamente establecida en el sistema.	Se almacenan los horarios manteniendo datos históricos. Se valida el choque de horarios para: docente, aula y grupo. Se visualizan los horarios elaborados por otros usuarios.
R11	Generar informes	Crear e imprimir reportes con la información almacenada.	Se generan reportes con la información actual del sistema. Se almacenan formatos de reporte para su posterior uso.

<b>R12</b>	Control de usuarios	<p>Almacenar, recuperar y editar datos de usuarios y permisos.</p> <p>Restringir el acceso al sistema basado en los permisos establecidos.</p>	<p>Se valida el acceso al sistema mediante usuario y clave secreta.</p> <p>Se permite la creación de horarios solamente a los usuarios autorizados.</p> <p>Se agrupan permisos en roles, para su uso independiente.</p>
------------	---------------------	--	---

### 3.6 Detalle de requerimientos

<b>R1</b>	<b>Registro de facultades</b>
<b>Descripción</b>	Se registran las facultades pertenecientes a la universidad, las cuales se utilizan para agrupar las carreras.
<b>Entradas</b>	Nombre, Comentario, Estado, Siglas.
<b>Responsable</b>	Coordinadores, Administrador de sistema.
<b>Acciones</b>	Guardar, Editar, Cambiar estado.
<b>Restricciones</b>	No debe haber campos vacíos (exceptuando comentarios).

<b>R2</b>	<b>Registro de aulas</b>
<b>Descripción</b>	Se registran los datos de las aulas existentes, las cuales serán asignadas a diferentes grupos.

<b>Entradas</b>	Nombre, Edificio, Medios audiovisuales, Comentario, Estado.
<b>Responsable</b>	Coordinadores, Administrador de sistema.
<b>Acciones</b>	Guardar, Editar, Cambiar estado.
<b>Restricciones</b>	No debe haber campos vacíos (exceptuando comentarios).

<b>R3</b>	<b>Registro de asignaturas</b>
<b>Descripción</b>	Se registran las asignaturas, se asocian a las distintas carreras y a diferentes docentes.
<b>Entradas</b>	Nombre, Comentario, Estado, Tipo, Carreras asociadas, Docentes asociados.
<b>Responsable</b>	Coordinadores, Administrador de sistema.
<b>Acciones</b>	Guardar, Editar, Cambiar estado.
<b>Restricciones</b>	Nombre, estado y tipo son obligatorios.

<b>R4</b>	<b>Registro de docentes</b>
<b>Descripción</b>	Se registran los datos de los docentes que impartirán las asignaturas de cada carrera.
<b>Entradas</b>	Nombre, Comentario, Estado, Tipo, Apellido, Asignaturas asociadas, Teléfonos, Correos, Dirección, Ciudad, Título.
<b>Responsable</b>	Coordinadores, Administrador de sistema.

<b>Acciones</b>	Guardar, Editar, Cambiar estado.
<b>Restricciones</b>	Nombre, apellido, estado y tipo son obligatorios.

<b>R5</b>	<b>Registro de grupos</b>
<b>Descripción</b>	Se registran los grupos de cada periodo para las distintas carreras.
<b>Entradas</b>	Nombre, Comentario, Año, Año lectivo, Turno, Carrera, Aula.
<b>Responsable</b>	Coordinadores, Administrador de sistema.
<b>Acciones</b>	Guardar, Editar.
<b>Restricciones</b>	No debe haber campos vacíos (exceptuando comentarios). El año debe estar comprendido entre 1 y 5.

<b>R6</b>	<b>Registro de carreras</b>
<b>Descripción</b>	Se registran las carreras de las distintas facultades.
<b>Entradas</b>	Nombre, Comentario, Estado, Facultad, Coordinador asociado, Duración, Pensum, Grupos.
<b>Responsable</b>	Coordinadores, Administrador de sistema.
<b>Acciones</b>	Guardar, Editar, Cambiar estado.



<b>Restricciones</b>	<p>Nombre, facultad, estado y duración son obligatorios.</p> <p>La duración debe ser un número entero positivo (inicialmente 10).</p>
----------------------	---

<b>R7</b>	<b>Registro de períodos</b>
<b>Descripción</b>	Se registran los periodos (semestres y trimestres) en que se impartirán los cursos.
<b>Entradas</b>	Nombre, Comentario, Estado, Año lectivo, Tipo, Fecha de inicio, Posición.
<b>Responsable</b>	Coordinadores, Administrador de sistema.
<b>Acciones</b>	Guardar, Editar, Cambiar estado.
<b>Restricciones</b>	<p>No debe haber campos vacíos (exceptuando comentarios).</p> <p>La posición está limitada por el tipo (si es semestre, 1-2; si es trimestre 1-4)</p>

<b>R8</b>	<b>Registro de turnos</b>
<b>Descripción</b>	Se registran los datos de los turnos en que se impartirán los cursos.
<b>Entradas</b>	Nombre, Comentario, Estado, Días, Horas.
<b>Responsable</b>	Coordinadores, Administrador de sistema.

<b>Acciones</b>	Guardar, Editar, Cambiar estado.
<b>Restricciones</b>	No debe haber campos vacíos (exceptuando comentarios).

<b>R9</b>	<b>Registro de pensum</b>
<b>Descripción</b>	Se registran los datos de que asignaturas se imparten en cada carrera.
<b>Entradas</b>	Carrera, Asignatura, Semestre, Duración.
<b>Responsable</b>	Coordinadores, Administrador de sistema.
<b>Acciones</b>	Guardar, Editar, Eliminar.
<b>Restricciones</b>	No debe haber campos vacíos. La duración debe ser un numero entero positivo.

<b>R10</b>	<b>Formar horarios</b>
<b>Descripción</b>	Se registran los datos de cada asignación de clases asociadas a grupos y docentes.
<b>Entradas</b>	Grupo, Asignatura, Docente, Periodo, Día, Hora de inicio, Hora de finalización, Usuario de creación, Estado.
<b>Responsable</b>	Coordinadores.
<b>Acciones</b>	Guardar, Editar, Solicitar intercambio.
<b>Restricciones</b>	No debe haber campos vacíos.

	Si hay un choque en docente o grupo se guardará con el estado "En conflicto". Los intercambios de horas deben ser autorizados por los usuarios de creación de los horarios afectados.
--	---

<b>R11</b>	<b>Generar informes</b>
<b>Descripción</b>	Se crean reportes con la información almacenada en el sistema.
<b>Entradas</b>	Selección del reporte deseado.
<b>Responsable</b>	Coordinadores, Administrador de sistema, Sub director.
<b>Acciones</b>	Guardar, Imprimir.
<b>Restricciones</b>	

<b>R12</b>	<b>Control de usuarios</b>
<b>Descripción</b>	Cada usuario del sistema deberá ser registrado previamente por el administrador, y tendrá acceso limitado al sistema.
<b>Entradas</b>	Nombre, Roles y permisos.
<b>Responsable</b>	Administrador de sistema.
<b>Acciones</b>	Guardar, Editar, Cambiar estado, Generar clave.
<b>Restricciones</b>	No debe haber campos vacíos. Debe haber uno o más roles asignados.

### **3.7 Requerimientos no funcionales**

#### **3.7.1 Portabilidad**

El sistema diseñado y sus componentes deben ser ejecutables en plataformas Windows (desde la versión XP), con máquinas que presentan arquitecturas de 32 y 64 bits. Se deben considerar las características actuales de infraestructura de la sede para la implementación de la solución. En la capa de datos el motor a utilizar debe ser ejecutable en plataformas Windows y GNU/Linux.

Debido a que la aplicación cliente se ejecutará solamente en la plataforma Windows y con el fin de aumentar el aislamiento y control sobre los archivos distribuibles deberá ser desarrollada como una aplicación de escritorio actualizable vía LAN.

#### **3.7.2 Flexibilidad**

La configuración de los parámetros de instalación no debe requerir modificaciones al código fuente de la instalación. Debe ser totalmente independiente de la topología de red utilizada, es decir, el sistema debe poder funcionar en múltiples esquemas de comunicación, tanto para equipos conectados remotamente, como para equipos conectados por una red LAN, WAN o Internet y todas las combinaciones anteriormente descritas

#### **3.7.3 Disponibilidad**

El sistema debe soportar una operación en alta disponibilidad, es decir, debe estar provisto de mecanismos o componentes que aseguren la continuidad del servicio y permitan el almacenamiento distribuido en múltiples servidores.

#### **3.7.4 Madurez**

Se requiere la utilización de componentes base o herramientas utilizadas para el diseño, construcción, pruebas e implementación reconocidas que tengan más de 3 años en el mercado, que tengan soporte por parte del fabricante, que exista un fabricante reconocido y con trayectoria y que exista el desarrollo continuo de cada herramienta que permita el mejoramiento y acceso a nuevas versiones de acuerdo con la evolución de las plataformas.

#### **3.7.5 Concurrencia**

El sistema debe ser capaz de manejar múltiples usuarios conectados concurrentemente. Se requiere operar adecuadamente con hasta 15 sesiones concurrentes.

#### **3.7.6 Interface de usuario**

Los formularios y demás herramientas de apoyo deben ser intuitivos al usuario, su despliegue frente al usuario debe ser rápida, permitir su navegación a través de menús sencillos, autoajustable a cualquier tamaño y resolución de pantalla del usuario, utilizar imágenes optimizadas y componentes de diseño que permitan mostrar la información de manera dinámica, ágil y estética

#### **3.7.7 Desempeño**

Los tiempos de respuesta relacionados con formularios de manejo de información adición, modificación, eliminación, consulta de registros, autenticación y emisión de avisos y confirmaciones por parte del usuario, en forma general, no debe ser superior a 2.5 segundos, los informes y consultas que presenten una complejidad mediana no deberá exceder el tiempo de 4 segundos.

## Capítulo 4. Proceso de iteraciones Scrum

### 4.1 Historial de revisiones

Fecha	Versión	Descripción	Autores
28/09/2015	0.1	Gestión de infraestructura: pabellones, aulas y medios audiovisuales asociados.	Br. Sergio Velásquez Br. Justo Rodríguez
05/10/2015	0.2	Gestión de docentes: datos personales, títulos, lugares de procedencia y categorías de docentes.	Br. Sergio Velásquez Br. Justo Rodríguez
12/10/2015	0.3	Gestión de tiempos: periodos, tipos de periodos, turnos, días, horas.	Br. Sergio Velásquez Br. Justo Rodríguez
19/10/2015	0.4	Gestión de pensum: facultades, carreras, asignaturas y unión de carrera-asignatura-semester.	Br. Sergio Velásquez Br. Justo Rodríguez
26/10/2015	0.5	Gestión de horarios: grupos, asignaciones de clases, solicitudes de intercambio y mensajes entre coordinadores.	Br. Sergio Velásquez Br. Justo Rodríguez
23/11/2015	0.6	Gestión de informes: reportes y análisis de datos.	Br. Sergio Velásquez Br. Justo Rodríguez
02/12/2015	0.7	Gestión de usuarios: usuarios, roles y permisos.	Br. Sergio Velásquez Br. Justo Rodríguez
16/12/2015	1.0	Integración general de todas las revisiones.	Br. Sergio Velásquez Br. Justo Rodríguez

## 4.2 Roles

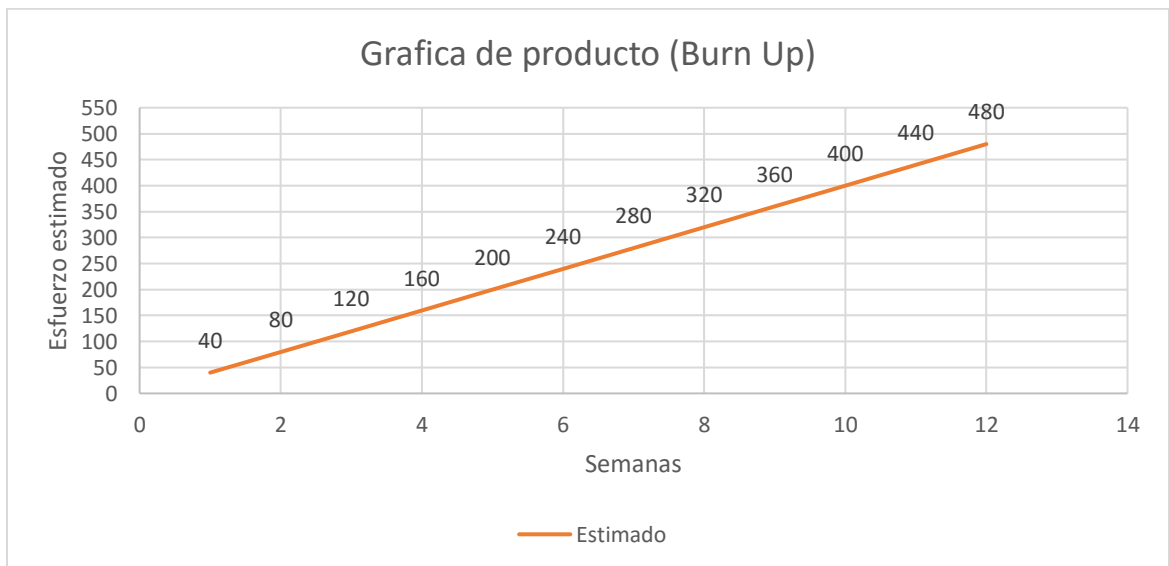
Implicados	Contacto	Rol
Ing. José Manuel Poveda	<a href="mailto:impov441@gmail.com">impov441@gmail.com</a>	Scrum Owner
Br. Sergio Velasquez	<a href="mailto:ing.svelasquez@gmail.com">ing.svelasquez@gmail.com</a>	Scrum Manager
Br. Justo Rodriguez	<a href="mailto:ing.leonelrv@gmail.com">ing.leonelrv@gmail.com</a>	Team
Br. Sergio Velasquez	<a href="mailto:ing.svelasquez@gmail.com">ing.svelasquez@gmail.com</a>	
Coordinadores de carrera	-	Stakeholders
M.Sc. Luis Dicovski	<a href="mailto:luisdi2005@yahoo.com">luisdi2005@yahoo.com</a>	

## 4.3 Product Backlog

ID	Prioridad	Título	Esfuerzo
001	Media	Registro de pabellones	10
002	Alta	Registro de aulas	13
003	Baja	Registro de medios audiovisuales	10
004	Baja	Asociación aulas y medios audiovisuales	7
005	Alta	Registro de docentes	20
006	Baja	Registro de ciudades	10
007	Media	Registro de categorías de docentes	10
008	Alta	Registro de períodos	8
009	Media	Registro de tipos de períodos	7
010	Alta	Registro de turnos	8
011	Alta	Registro de horas	5
012	Alta	Asociación entre turnos - días - horas	12
013	Baja	Registro de facultades	5
014	Alta	Registro de carreras	10
015	Alta	Registro de asignaturas	5
016	Alta	Creación de pensum (carrera - asignatura - semestre)	20

017	Alta	Registro de grupos	25
018	Muy Alta	Creación de horarios	100
019	Media	Peticiones de intercambio	35
020	Media	Creación de reportes	24
021	Baja	Creación de tablas de análisis	32
022	Alta	Creación de usuarios	15
023	Alta	Creación de roles	20
024	Media	Permisos por rol	45
025	Media	Integración general	24

#### 4.4 Gráfica de producto



El tiempo total de desarrollo es de 12 semanas calendario, correspondiente a 480 puntos de esfuerzos, donde:

- Cada punto de esfuerzo corresponde a una hora de trabajo efectivo.
- Los puntos de esfuerzos definidos abarcan la totalidad de trabajo a realizar. Incluyen, planificación, diseño, desarrollo, prueba y documentación.
- Se trabajará un total de 40 horas semanales.



## 4.5 Sprint #1 Gestión de infraestructura

### 4.5.1 Sprint Planning Meeting

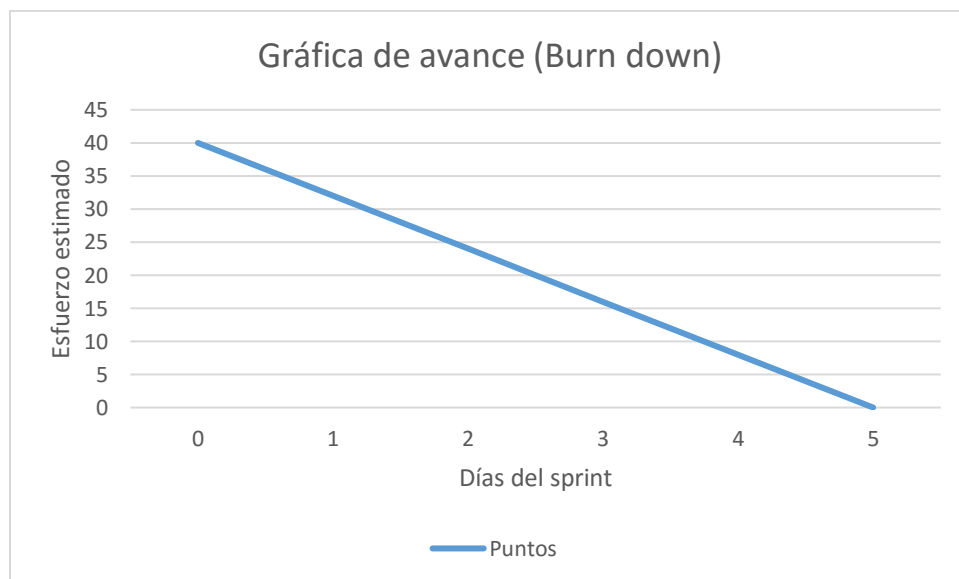
Se acuerda completar la funcionalidad relacionada al registro de la infraestructura del recinto, estos requerimientos se encuentran establecidos en los elementos 1, 2, 3 y 4 del Product Backlog, lo que abarca: pabellones, aulas y medios audiovisuales.

Se realizarán los modelos de base de datos, clases de lógica de negocio, interfaz gráfica de usuario y pruebas de validación, lo que en su conjunto suman un total de 40 puntos de esfuerzo.

### 4.5.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas, no se encontraron problemas destacables durante el proceso.

### 4.5.3 Burn down chart



#### 4.5.4 Sprint Backlog

			Proyecto			28-sep	29-sep	30-sep	01-oct	02-oct					
			Inicio	Fin	Jornada										
			28/09/2015	02/10/2015	40hrs										
			Tareas pendientes								7	6	4	3	1
			Horas de trabajo pendientes								40	32	24	16	8
Pila del sprint					Esfuerzo										
Categoría	Tarea	Responsable	Estimado en horas	Estado											
Diseño	Crear tablas base de datos	TEAM	8	Completo	6	1	1								
Diseño	Validar diseño de base de datos	TEAM	4	Completo	2	1	1								
Desarrollo	Programar clases de lógica	TEAM	8	Completo		6	1	1							
Desarrollo	Crear y programar interfaz para Edificios	TEAM	4	Completo			2	2							
Desarrollo	Crear y programar interfaz para Aulas	TEAM	4	Completo			3	1							
Desarrollo	Crear y programar interfaz para Medios	TEAM	4	Completo				4							
Prueba	Validar restricciones	TEAM	8	Completo					8						

## 4.6 Sprint #2 Gestión de docentes

### 4.6.1 Sprint Planning Meeting

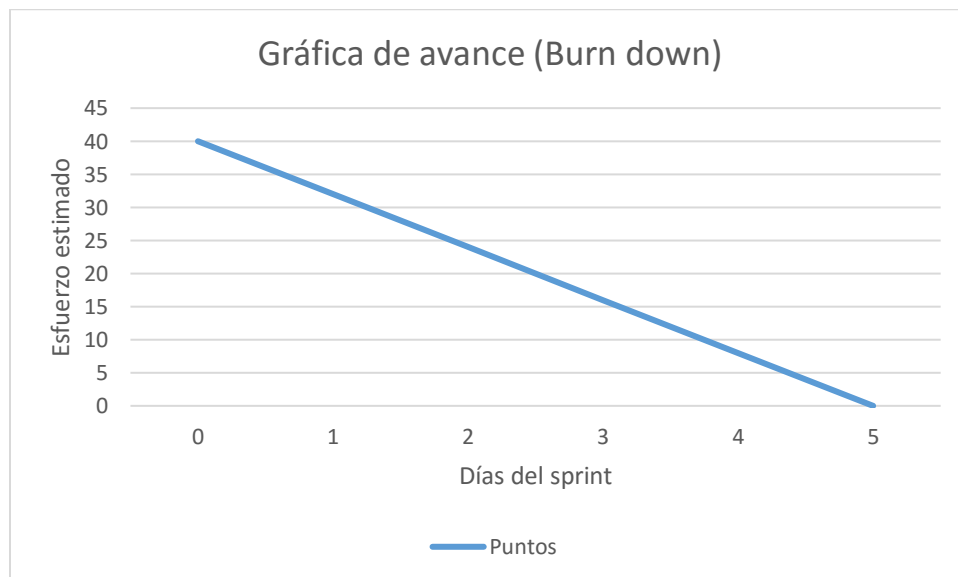
Se acuerda completar la funcionalidad relacionada al registro los docentes, estos requerimientos se encuentran establecidos en los elementos 5, 6, y 7 del Product Backlog, lo que abarca: docentes, tipos de docentes y ciudades de procedencia.

Se realizarán los modelos de base de datos, clases de lógica de negocio, interfaz gráfica de usuario y pruebas de validación, lo que en su conjunto suman un total de 40 puntos de esfuerzo.

### 4.6.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas, no se encontraron problemas destacables durante el proceso.

### 4.6.3 Burn down chart



#### 4.6.4 Sprint Backlog

					Proyecto			05-sep	06-sep	07-sep	08-sep	09-sep
					Inicio	Fin	Jornada					
					05/10/2015	09/10/2015	40hrs					
					Tareas pendientes							
					Horas de trabajo pendientes							
					5	4	3	2	1			
					40	32	24	16	8			
Pila del sprint					Esfuerzo							
Categoría	Tarea	Responsable	Estimado en horas	Estado								
Diseño	Crear tablas base de datos	TEAM	8	Completo	6	1	1					
Diseño	Validar diseño de base de datos	TEAM	4	Completo	2	1	1					
Desarrollo	Programar clases de logica	TEAM	8	Completo		6	1	1				
Desarrollo	Crear y programar interfaz para Docentes	TEAM	8	Completo			5	3				
Desarrollo	Crear y programar interfaz para Tipo Docente	TEAM	2	Completo				2				
Desarrollo	Crear y programar interfaz para ciudad	TEAM	2	Completo				2				
Prueba	Validar restricciones	TEAM	8	Completo					8			

## 4.7 Sprint #3 Gestión de tiempo

### 4.7.1 Sprint Planning Meeting

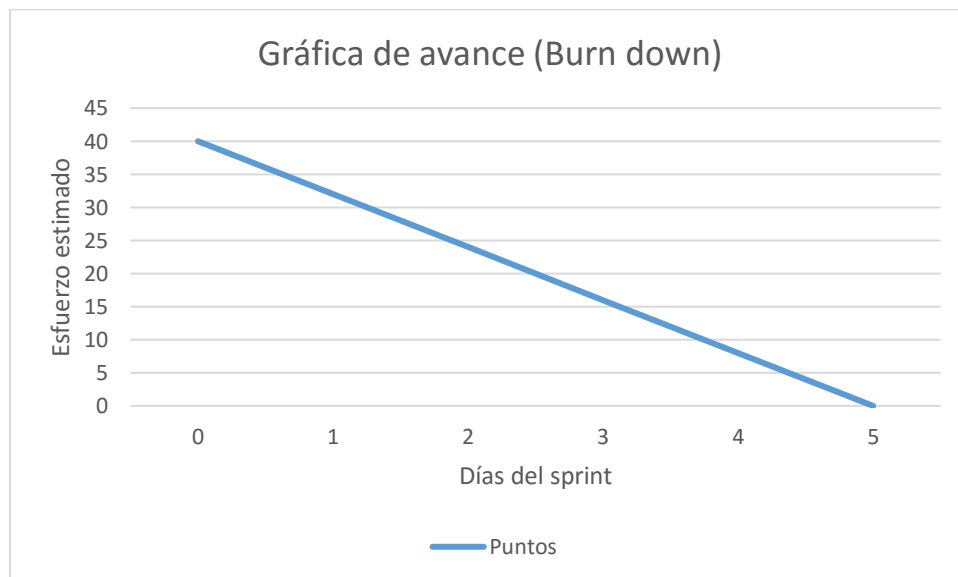
Se acuerda completar la funcionalidad relacionada al control de los tiempos, estos requerimientos se encuentran establecidos en los elementos 8, 9, 10, 11 y 12 del Product Backlog, lo que abarca: turnos, días, horas y períodos.

Se realizarán los modelos de base de datos, clases de lógica de negocio, interfaz gráfica de usuario y pruebas de validación, lo que en su conjunto suman un total de 40 puntos de esfuerzo.

### 4.7.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas, no se encontraron problemas destacables durante el proceso.

### 4.7.3 Burn down chart



#### 4.7.4 Sprint Backlog

					Proyecto			12-oct	13-oct	14-oct	15-oct	16-oct	
					Inicio		Fin						Jornada
					12/10/2015		16/10/2015						40hrs
					Tareas pendientes			6	5	3	2	1	
					Horas de trabajo pendientes			40	32	24	16	8	
Pila del sprint					Esfuerzo								
Categoría	Tarea	Responsable	Estimado en horas	Estado									
Diseño	Crear tablas base de datos	TEAM	8	Completo	6	1	1						
Diseño	Validar diseño de base de datos	TEAM	4	Completo	2	1	1						
Desarrollo	Programar clases de logica	TEAM	8	Completo		6	1	1					
Desarrollo	Crear y programar interfaz para Periodos	TEAM	6	Completo			5	1					
Desarrollo	Crear y programar interfaz para Turnos	TEAM	6	Completo				6					
Prueba	Validar restricciones	TEAM	8	Completo					8				

## 4.8 Sprint #4 Gestión de pensum

### 4.8.1 Sprint Planning Meeting

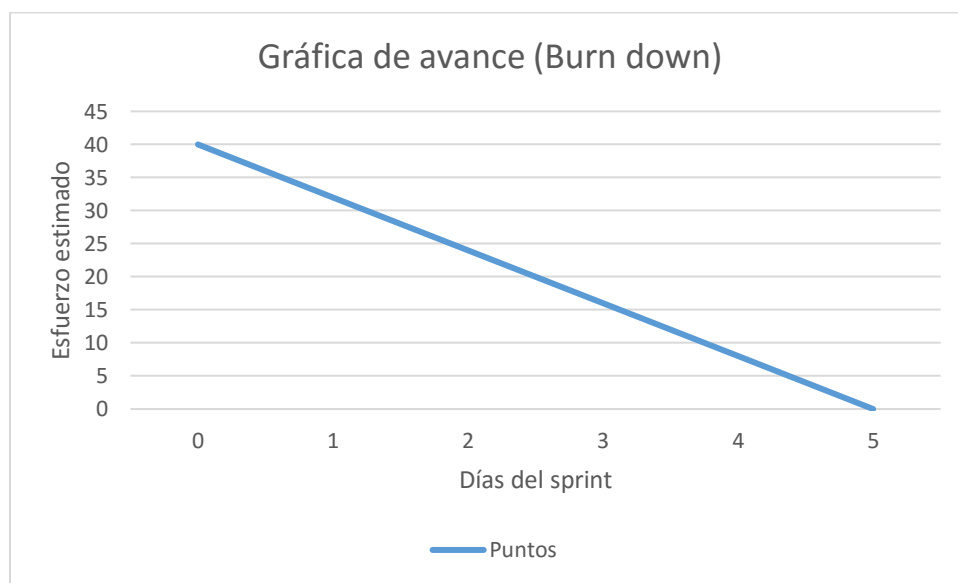
Se acuerda completar la funcionalidad relacionada a la creación del pensum, estos requerimientos se encuentran establecidos en los elementos 13, 14, 15, y 16 del Product Backlog, lo que abarca: facultad, carrera, asignatura y sus respectivas relaciones.

Se realizarán los modelos de base de datos, clases de lógica de negocio, interfaz gráfica de usuario y pruebas de validación, lo que en su conjunto suman un total de 40 puntos de esfuerzo.

### 4.8.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas, no se encontraron problemas destacables durante el proceso.

### 4.8.3 Burn down chart



#### 4.8.4 Sprint Backlog

			Proyecto			19-oct	20-oct	21-oct	22-oct	23-oct
			Inicio	Fin	Jornada					
			19/10/2015	23/10/2015	40hrs					
			Tareas pendientes			7	6	4	3	1
			Horas de trabajo pendientes			40	32	24	16	8
Pila del sprint						Esfuerzo				
Categoría	Tarea	Responsable	Estimado en horas	Estado						
Diseño	Crear tablas base de datos	TEAM	8	Completo	6	1	1			
Diseño	Validar diseño de base de datos	TEAM	4	Completo	2	1	1			
Desarrollo	Programar clases de lógica	TEAM	8	Completo		6	1	1		
Desarrollo	Crear y programar interfaz para Facultad y Carreras	TEAM	6	Completo			5	1		
Desarrollo	Crear y programar interfaz para Asignaturas	TEAM	6	Completo				6		
Prueba	Validar restricciones	TEAM	8	Completo					8	



## 4.9 Sprint #5 Gestión de horarios

### 4.9.1 Sprint Planning Meeting

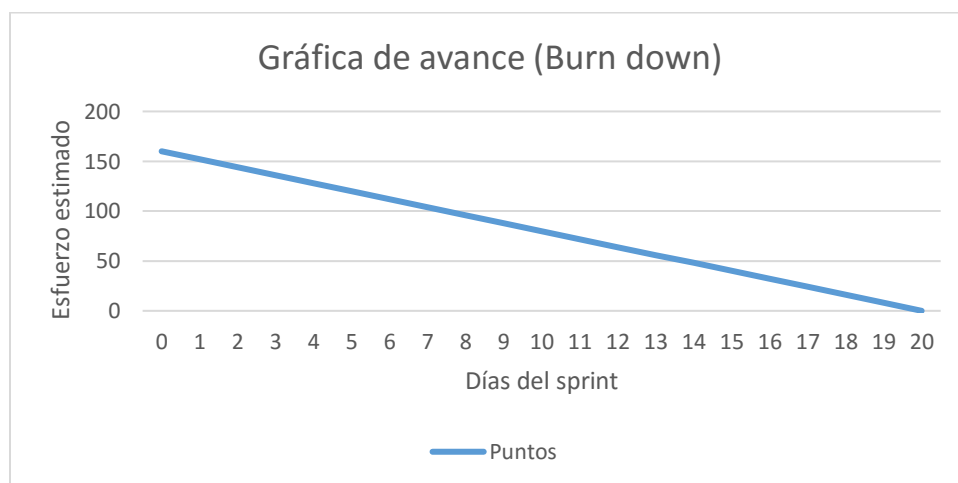
Se acuerda completar la funcionalidad relacionada a la creación de horarios, estos requerimientos se encuentran establecidos en los elementos 17, 18, y 19 del Product Backlog, lo que abarca: grupo, asignaciones, peticiones de intercambio y sus respectivas relaciones.

Se realizarán los modelos de base de datos, clases de lógica de negocio, interfaz gráfica de usuario y pruebas de validación, lo que en su conjunto suman un total de 160 puntos de esfuerzo.

### 4.9.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas. Se realizaron ajustes a la interfaz de creación de horarios para incluir vistas del calendario de docentes y grupos simultáneamente, sin embargo, esta tarea se llevó a cabo dentro del periodo de tiempo establecido, por lo que no representó un retraso a los tiempos globales.

### 4.9.3 Burn down chart



#### 4.9.4 Sprint Backlog

					Proyecto			26-oct	27-oct	28-oct	29-oct	30-oct	02-nov	03-nov	04-nov	05-nov	06-nov	09-nov	10-nov	11-nov	12-nov	13-nov	16-nov	17-nov	18-nov	19-nov	20-nov
					Inicio	Fin	Jornada																				
					26/10/2015	20/11/2015	160hrs																				
					Tareas pendientes																						
					Horas de trabajo pendientes																						
7					7	7	5	5	5	4	4	3	3	3	3	3	2	2	2	1	1	1	1	1	1		
160					152	144	144	136	128	120	112	104	96	96	88	88	72	64	56	48	40	40	40	40	32		
Pila del sprint					Esfuerzo																						
Categoría	Tarea	Responsable	Estimado en horas	Estado																							
Diseño	Crear tablas base de datos	TEAM	8	Completo	5	1	1	1																			
Diseño	Validar diseño de base de datos	TEAM	16	Completo	3	5	4	4																			
Desarrollo	Programar clases de lógica	TEAM	24	Completo		2	3	3	8	8																	
Desarrollo	Crear y programar interfaz para Grupos	TEAM	16	Completo							8	8															
Desarrollo	Crear y programar interfaz para Asignaciones	TEAM	40	Completo										8	8	8	8	8									
Desarrollo	Crear y programar interfaz para intercambios	TEAM	24	Completo															8	8	8						
Prueba	Validar restricciones	TEAM	32	Completo																			8	8	8	8	

## 4.10 Sprint #6 Gestión de informes

### 4.10.1 Sprint Planning Meeting

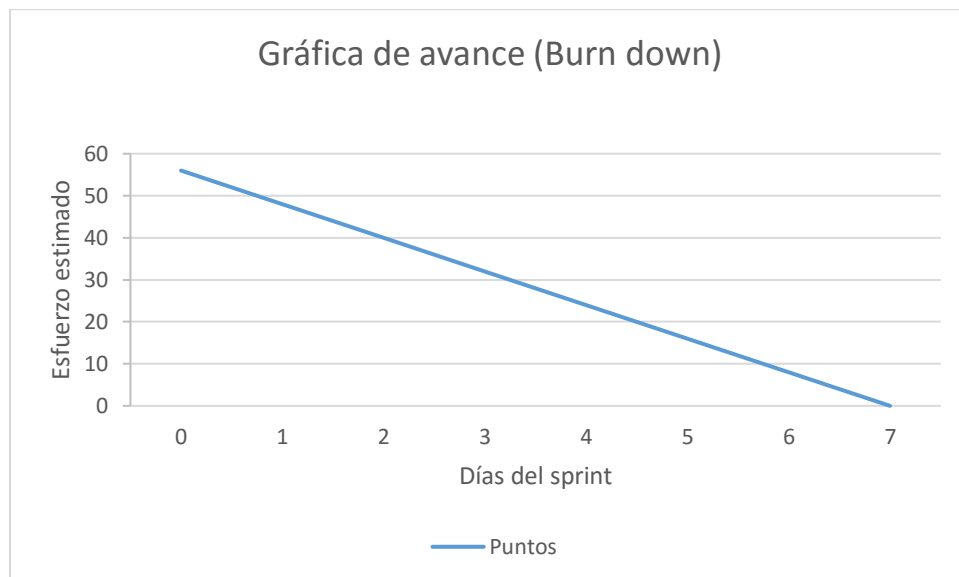
Se acuerda completar la funcionalidad relacionada a la creación e impresión de informes, estos requerimientos se encuentran establecidos en los elementos 20 y 21 del Product Backlog, lo que abarca: reportes y tablas de análisis.

Se realizarán los modelos de base de datos, clases de lógica de negocio, interfaz gráfica de usuario y pruebas de validación, lo que en su conjunto suman un total de 56 puntos de esfuerzo.

### 4.10.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas, no se encontraron problemas destacables durante el proceso.

### 4.10.3 Burn down chart



## 4.10.4 Sprint Backlog

					Proyecto			23-nov	24-nov	25-nov	26-nov	27-nov	30-nov	01-dic
					Inicio	Fin	Jornada							
					23/11/2015	01/12/2015	56hrs							
					Tareas pendientes			5	5	4	3	3	3	1
					Horas de trabajo pendientes			100	92	84	84	84	76	68
Pila del sprint					Esfuerzo									
Categoría	Tarea	Responsable	Estimado en horas	Estado										
Diseño	Crear y validar tabla en base de datos	TEAM	8	Completo	4	4								
Desarrollo	Programar clases de lógica	TEAM	16	Completo	4	4	8							
Desarrollo	Crear y programar interfaz para Análisis	TEAM	12	Completo				4	4	4				
Desarrollo	Crear y programar interfaz para Informes	TEAM	12	Completo				4	4	4				
Prueba	Validar restricciones	TEAM	8	Completo							8			

## 4.11 Sprint #7 Gestión de usuarios

### 4.11.1 Sprint Planning Meeting

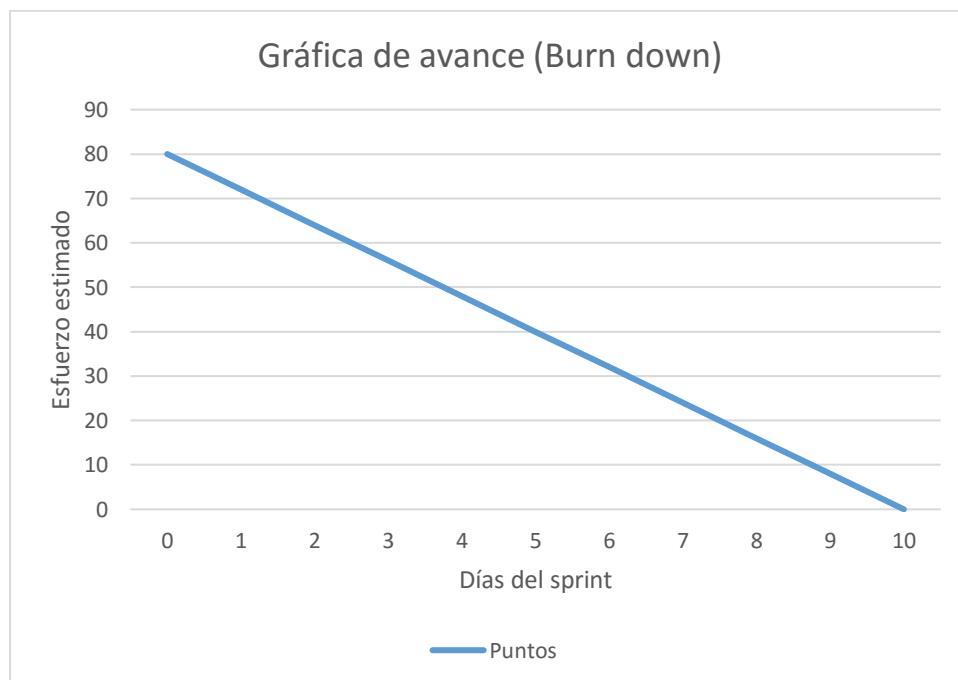
Se acuerda completar la funcionalidad relacionada a la gestión de usuarios, estos requerimientos se encuentran establecidos en los elementos 22, 23 y 24 del Product Backlog, lo que abarca: usuarios, roles y permisos.

Se realizarán los modelos de base de datos, clases de lógica de negocio, interfaz gráfica de usuario y pruebas de validación, lo que en su conjunto suman un total de 80 puntos de esfuerzo.

### 4.11.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas, no se encontraron problemas destacables durante el proceso.

### 4.11.3 Burn down chart



#### 4.11.4 Sprint Backlog

					Proyecto			02-dic	03-dic	04-dic	07-dic	08-dic	09-dic	10-dic	11-dic	14-dic	15-dic
					Inicio	Fin	Jornada										
					02/12/2015	15/12/2015	80hrs										
					Tareas pendientes			6	6	6	4	4	3	2	2	2	1
					Horas de trabajo pendientes			80	72	64	56	48	40	32	24	16	8
Pila del sprint																	
Categoría	Tarea	Responsable	Estimado en horas	Estado	Esfuerzo												
Diseño	Crear tablas base de datos	TEAM	16	Completo	6	6	4										
Diseño	Validar diseño de base de datos	TEAM	8	Completo	2	2	4										
Desarrollo	Programar clases de logica	TEAM	16	Completo				8	8								
Desarrollo	Crear y programar interfaz Usuarios	TEAM	8	Completo						8							
Desarrollo	Crear y programar interfaz Roles y Permisos	TEAM	20	Completo								8	8	4			
Desarrollo	Validar restricciones	TEAM	12	Completo											4	8	

## 4.12 Sprint #8 Integración general

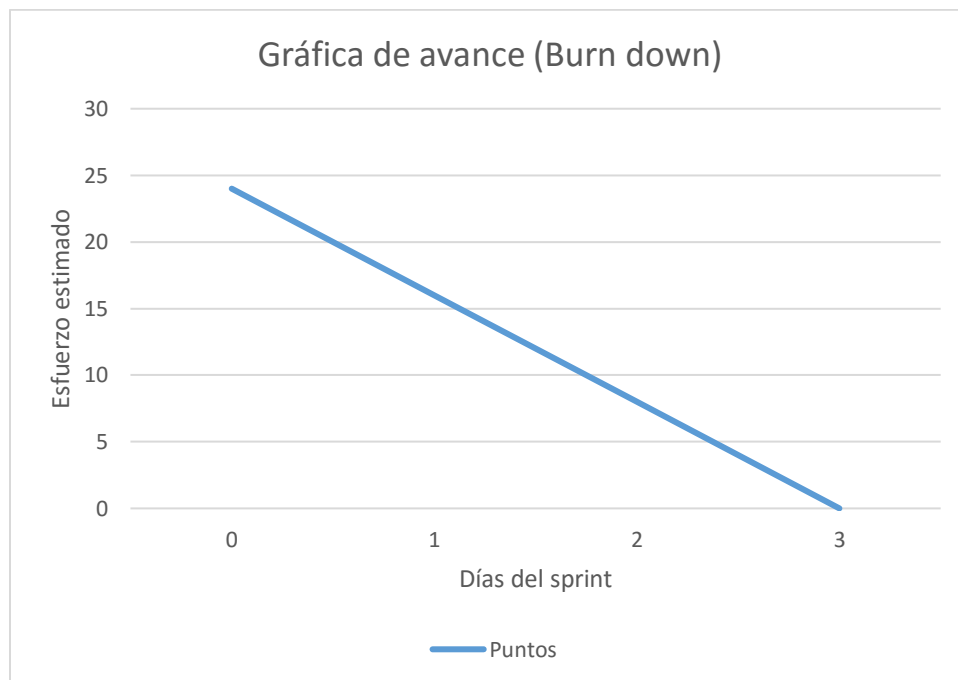
### 4.12.1 Sprint Planning Meeting

Se acuerda completar la integración de todos los elementos del sistema en un único software, este requerimiento se encuentra establecido en el elemento 25 del Product Backlog y suma un total de 24 puntos de esfuerzo.

### 4.12.2 Sprint Review Meeting

Se completó el 100% de las tareas programadas, no se encontraron problemas destacables durante el proceso.

### 4.12.3 Burn down chart



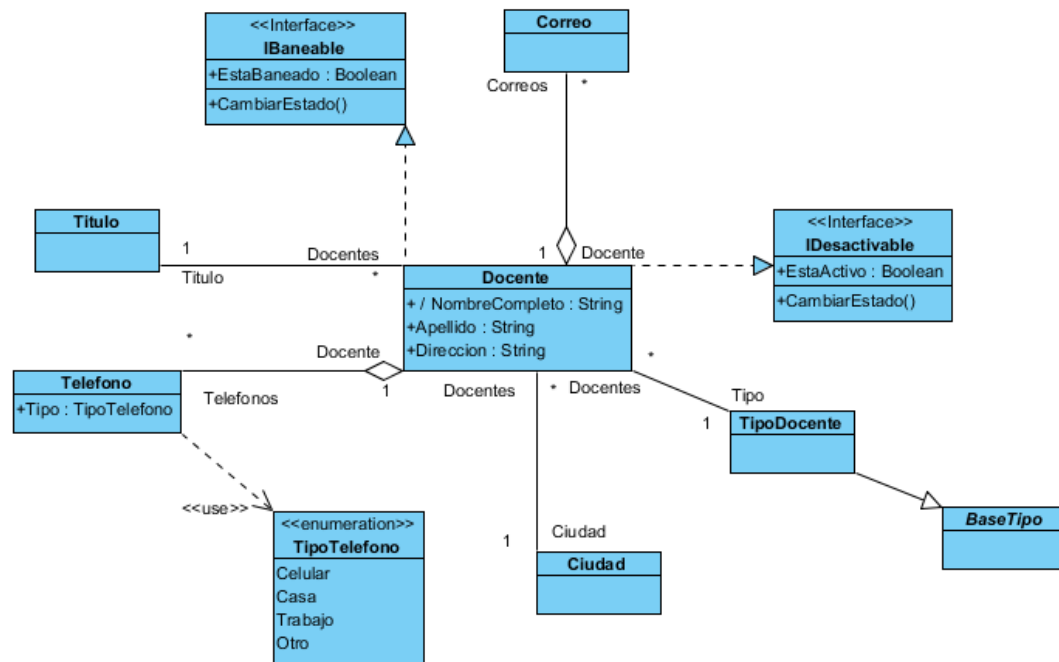
#### 4.12.4 Sprint Backlog

					Proyecto		
					Inicio	Fin	Jornada
					16/12/2015	18/12/2015	24hrs
					Tareas pendientes		
					Horas de trabajo pendientes		
					2	2	2
					24	16	8
Pila del sprint					Esfuerzo		
Categoría	Tarea	Responsable	Estimado en horas	Estado			
Desarrollo	Integrar módulos	TEAM	20	Completo	8	8	4
Prueba	Probar integración	TEAM	4	Completo			4

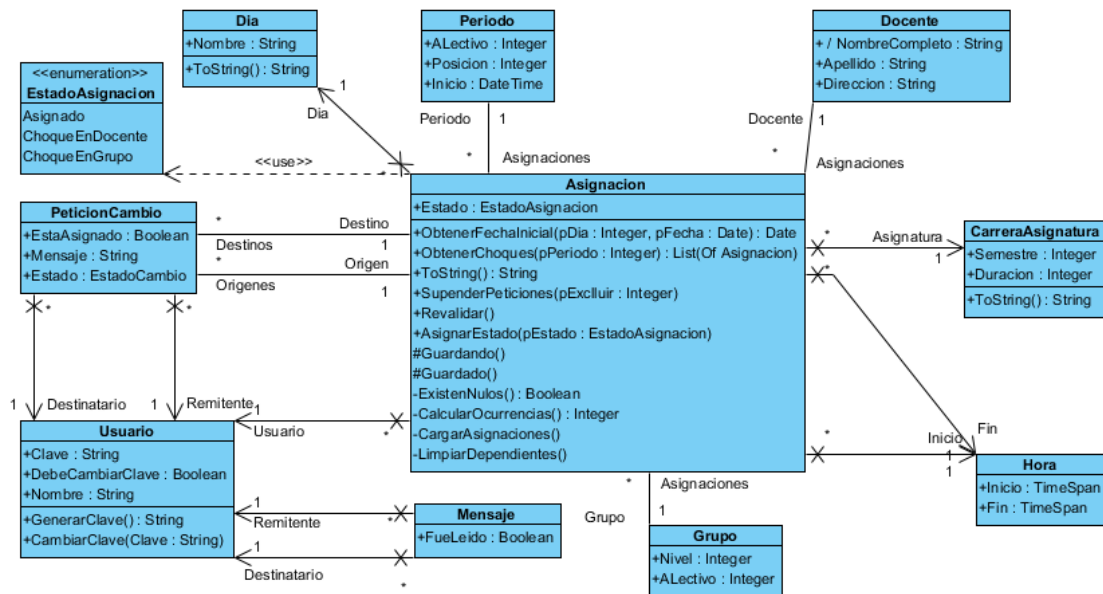




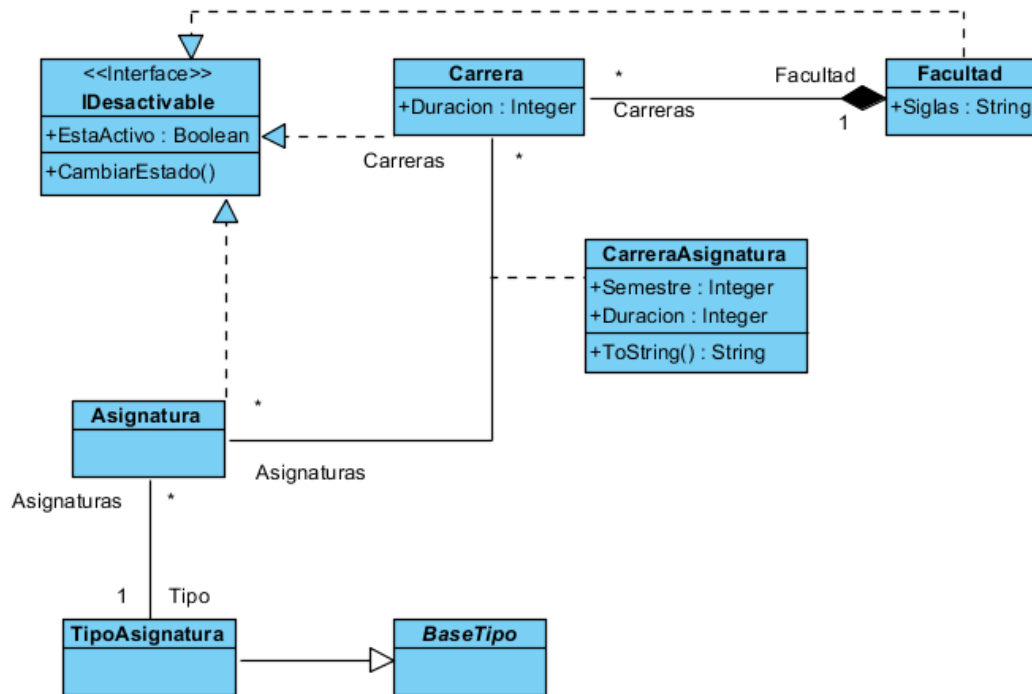
### 5.1.3 Docentes



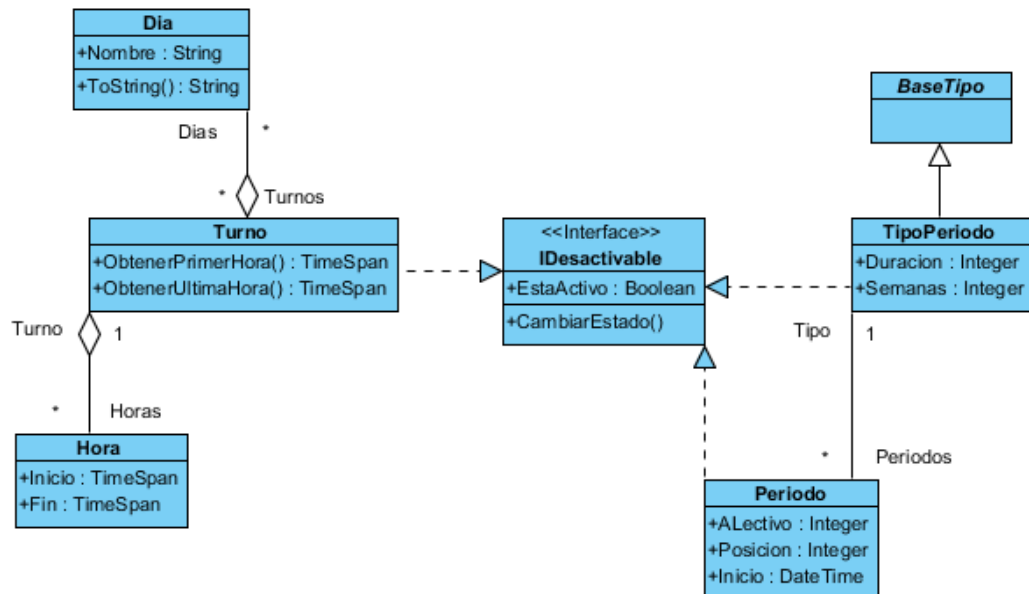
### 5.1.4 Horarios



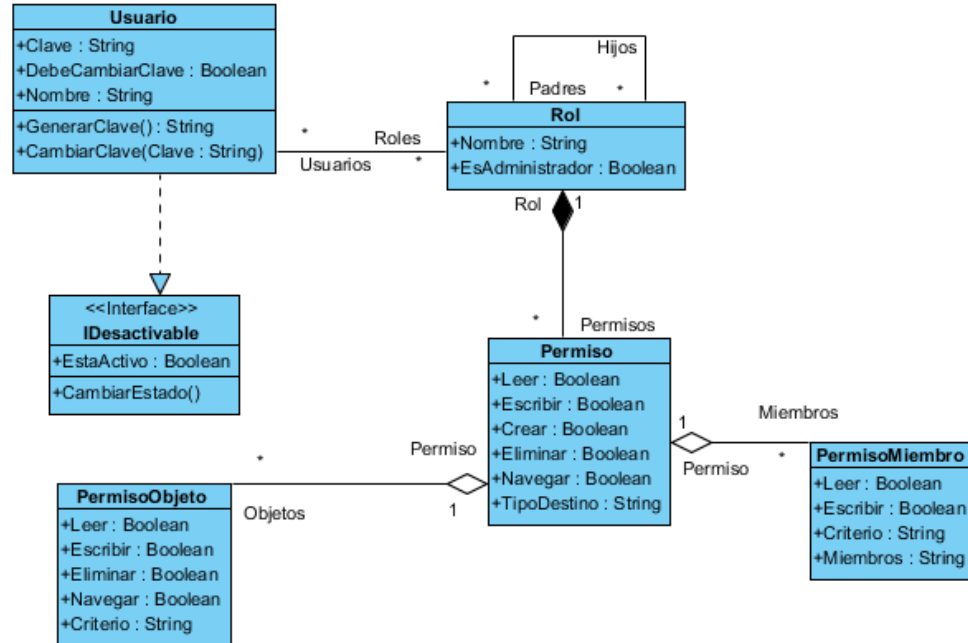
### 5.1.5 Pensum



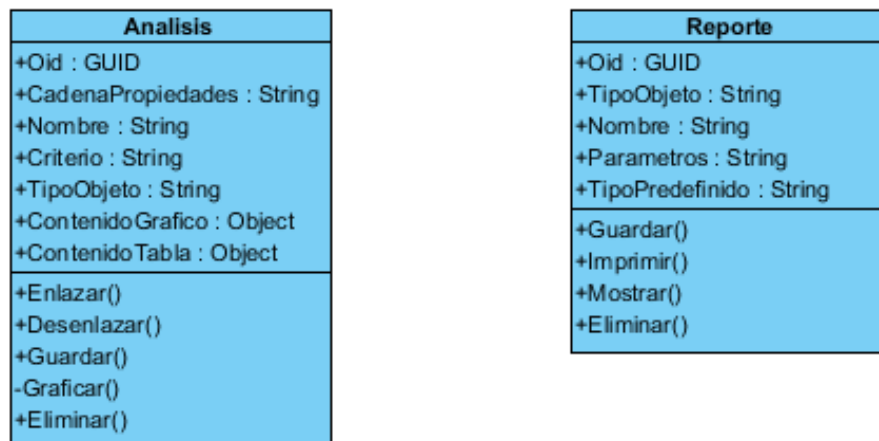
### 5.1.6 Tiempos



### 5.1.7 Usuarios



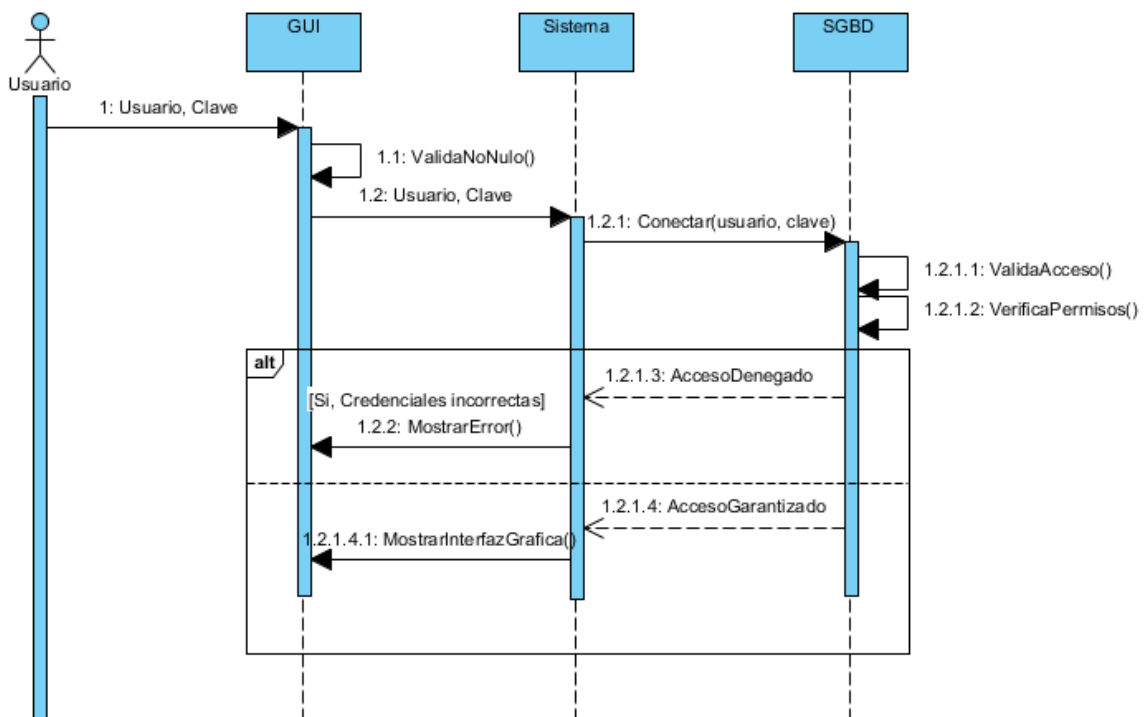
### 5.1.8 Reportes



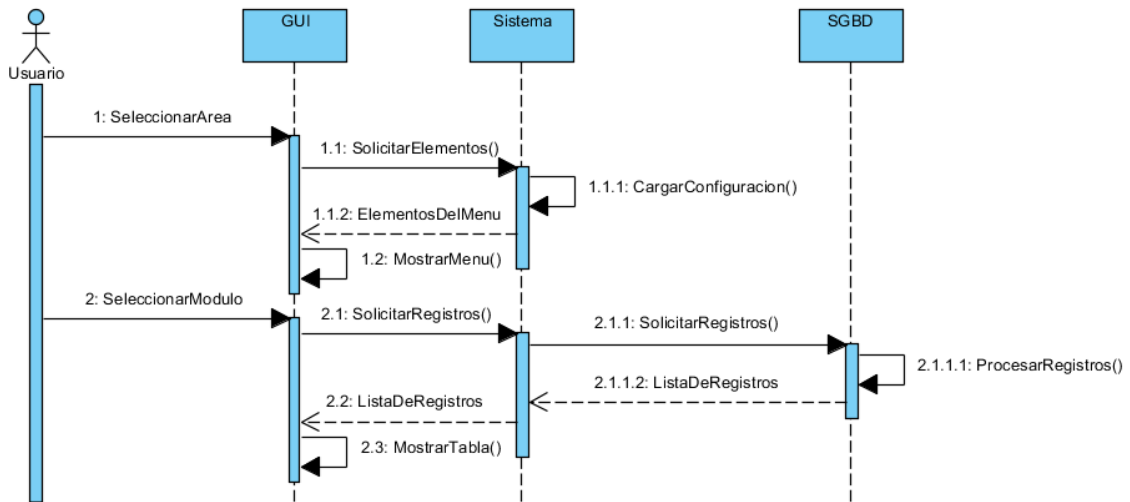
## 5.2 Diagramas de secuencia

Se muestran las interacciones, que representan la secuencia de mensajes entre diferentes componentes y actores del sistema a través del tiempo, se tomarán en cuenta solamente los escenarios y eventos generales, detallando la implementación, elementos y mensajes de cada caso.

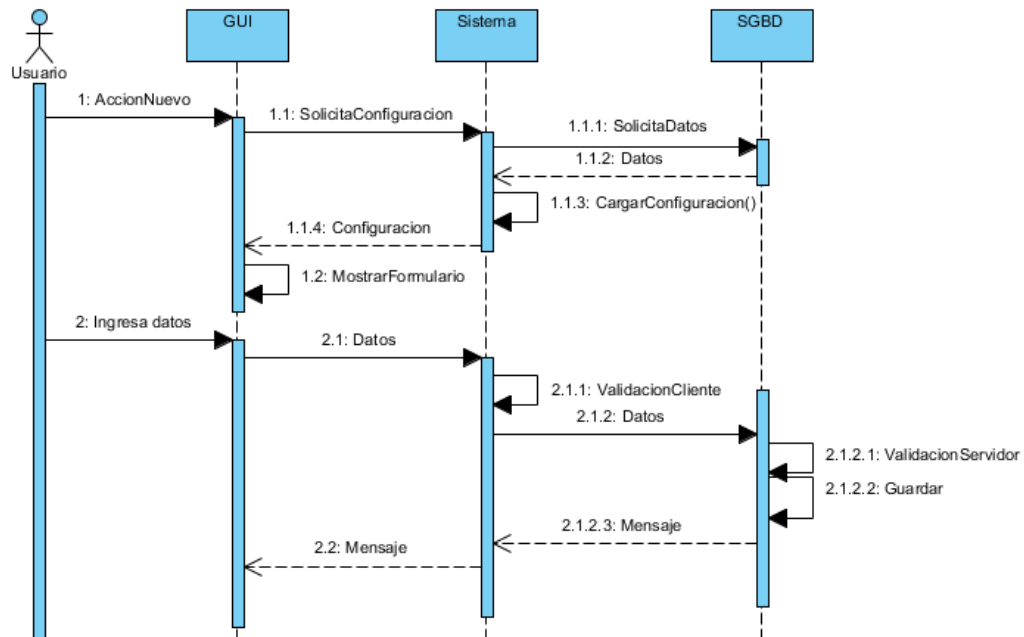
### 5.2.1 Inicio de sesion



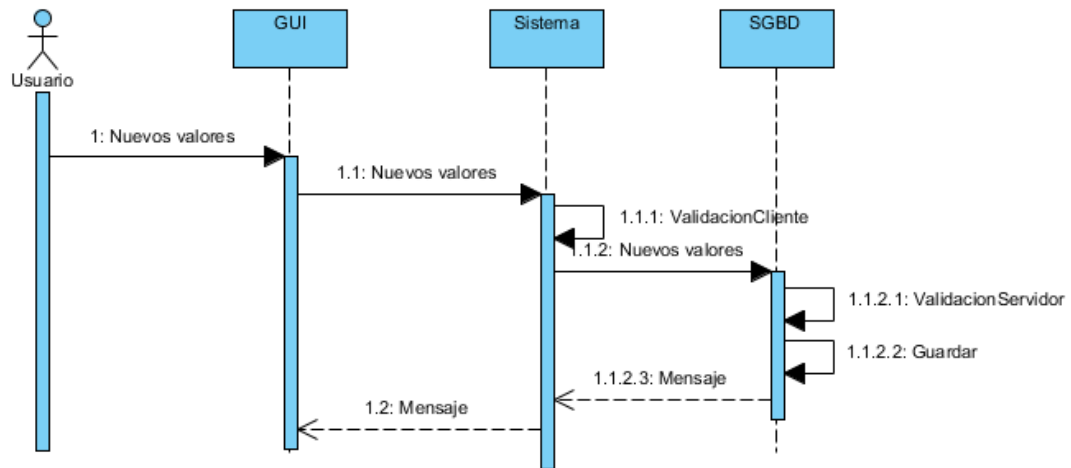
### 5.2.2 Navegar



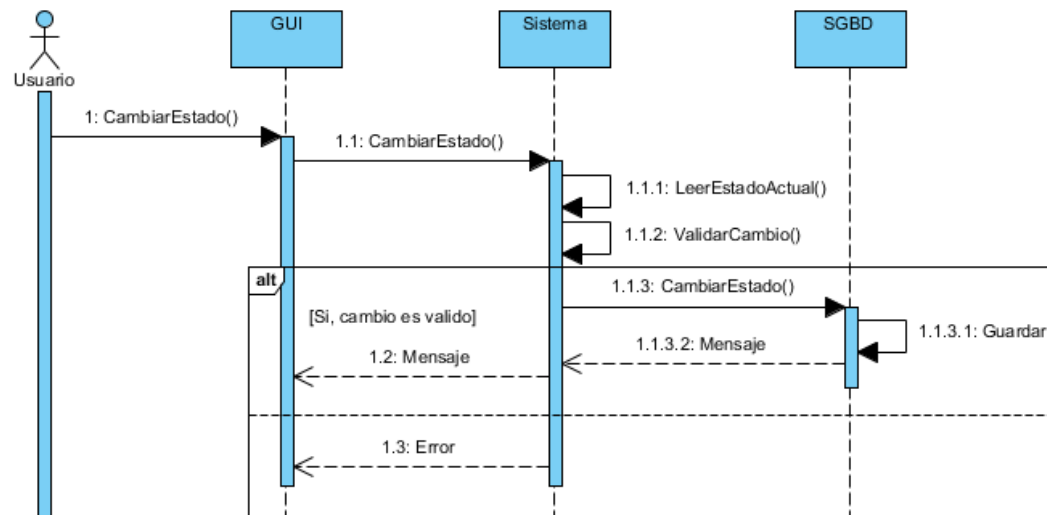
### 5.2.3 Crear nuevo elemento



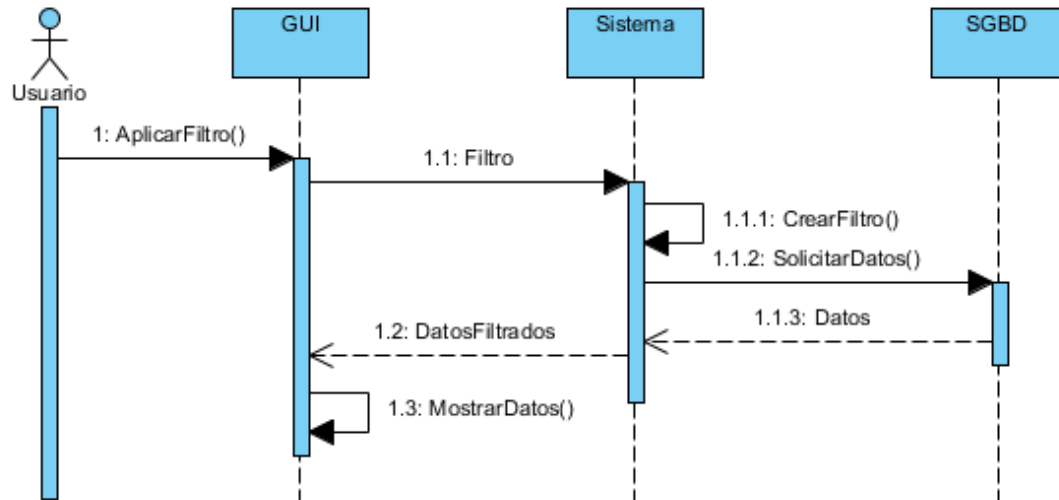
### 5.2.4 Editar elemento existente



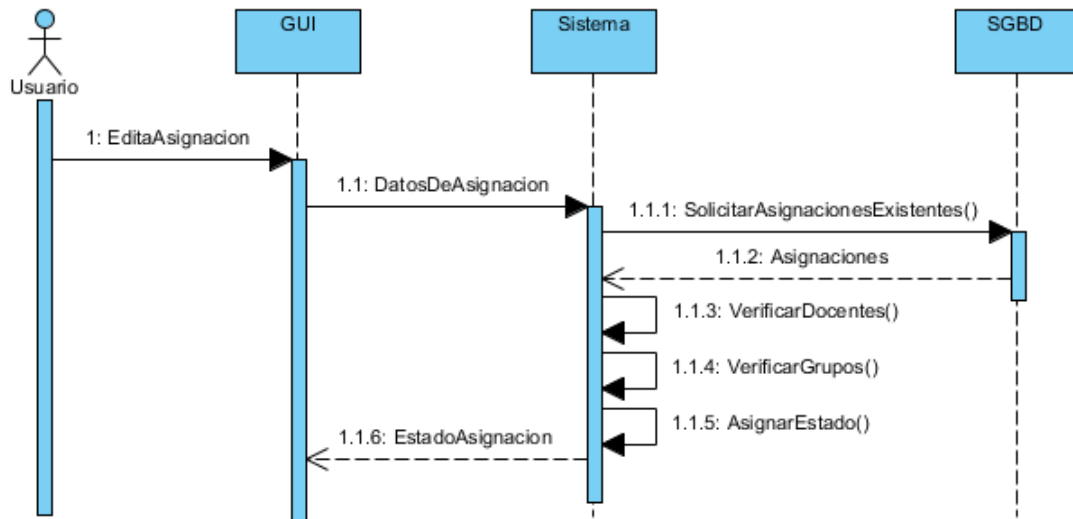
### 5.2.5 Cambiar estado



### 5.2.6 Filtrar horarios

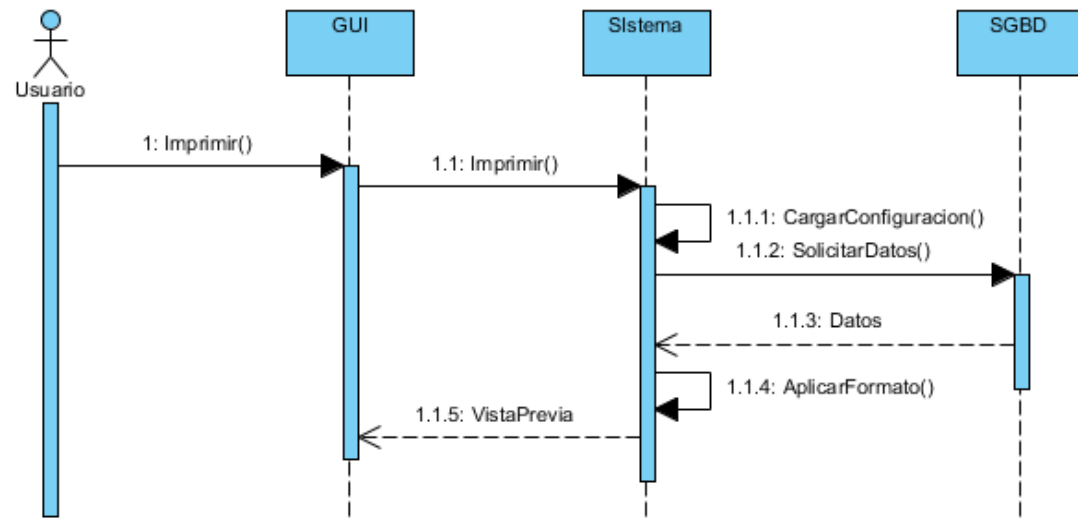


### 5.2.7 Validar choques

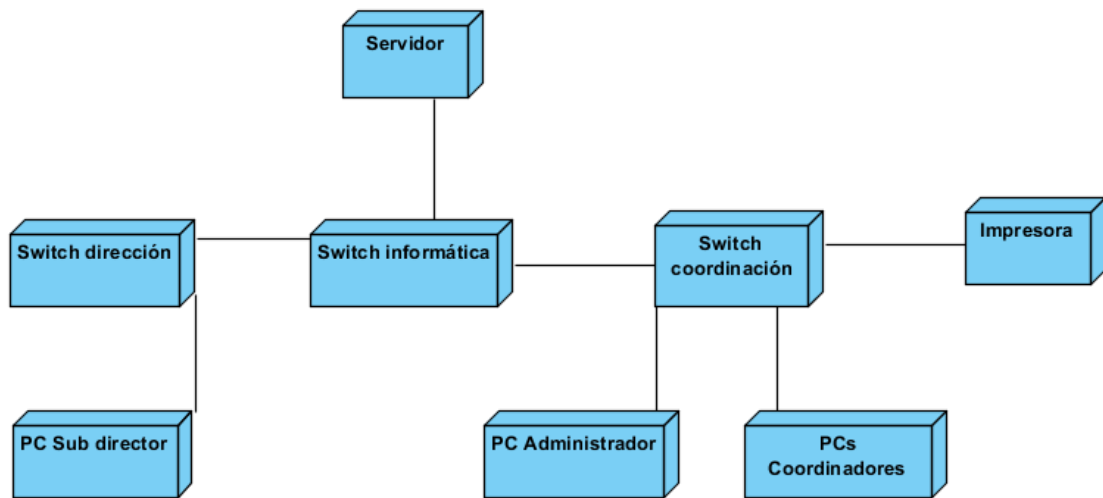




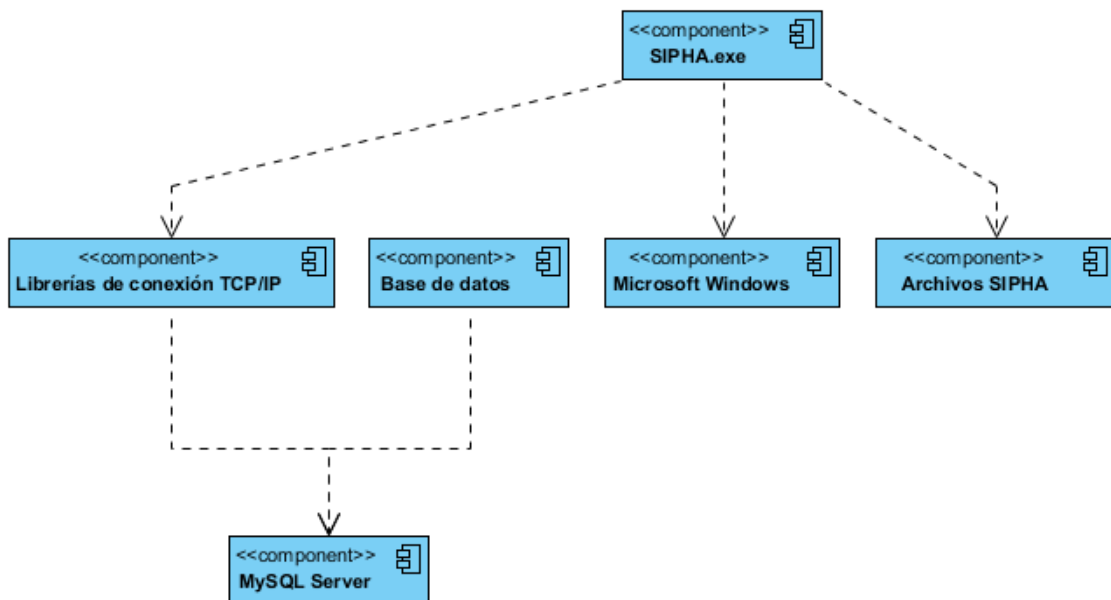
### 5.2.8 Imprimir reportes



### 5.3 Diagrama de distribución

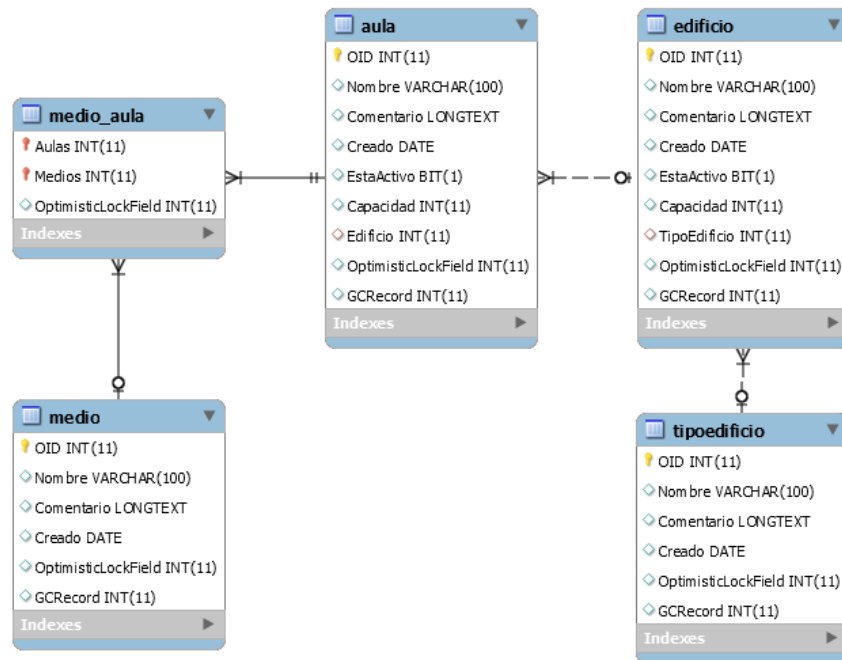


### 5.4 Diagrama de componentes

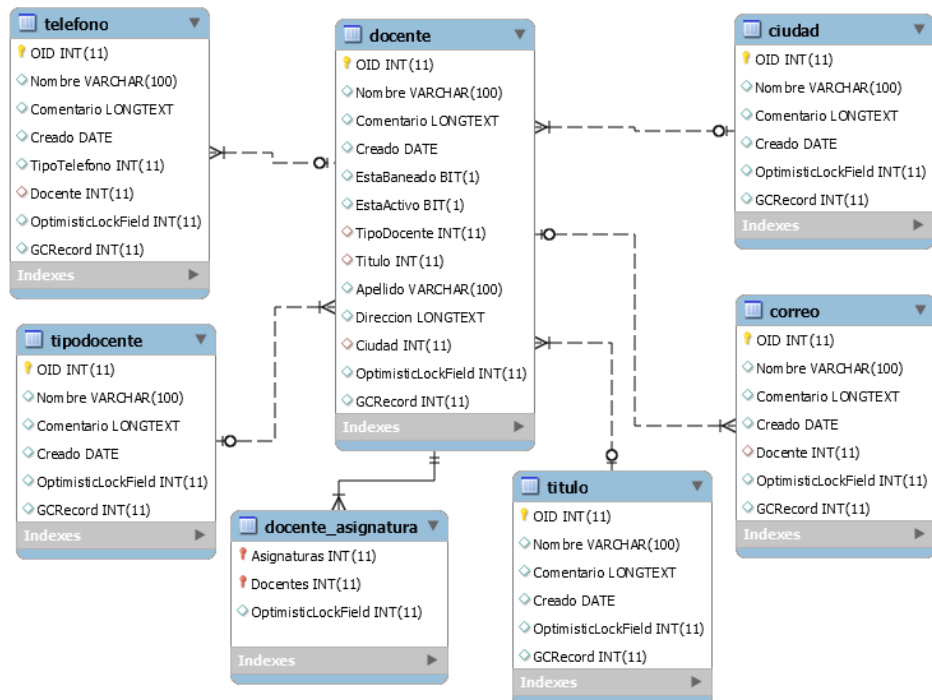


## Capítulo 6. Modelo de base de datos

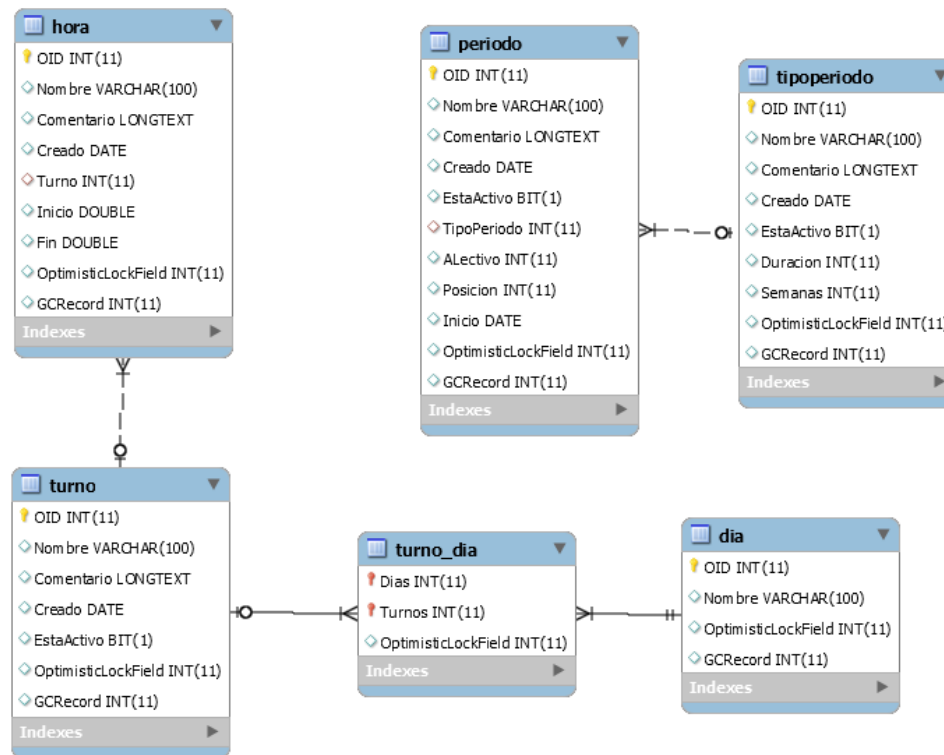
### 6.1 Modelo de infraestructura



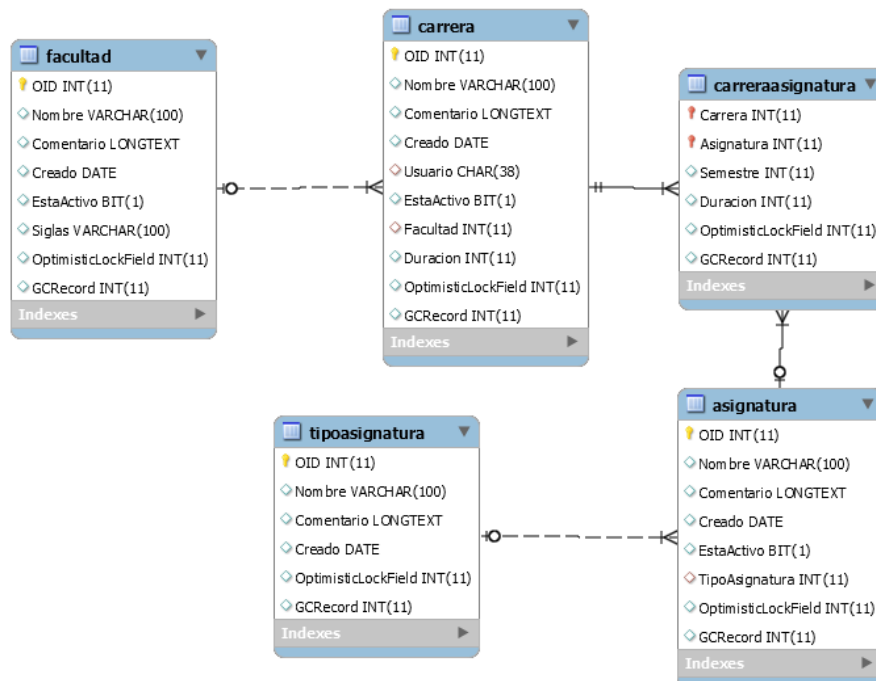
### 6.2 Modelo de docentes



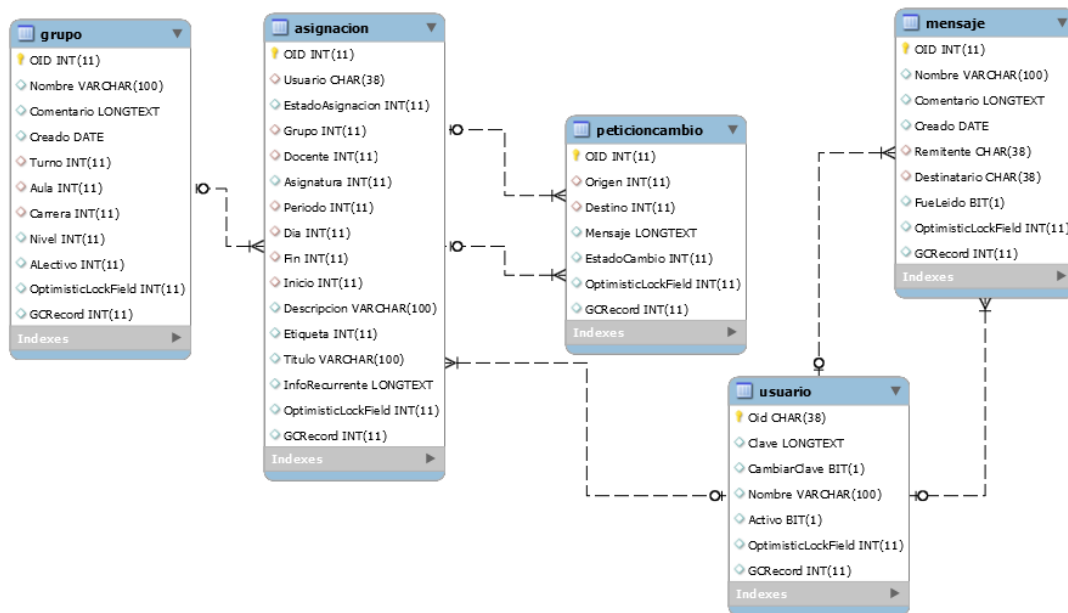
## 6.3 Modelo de tiempos



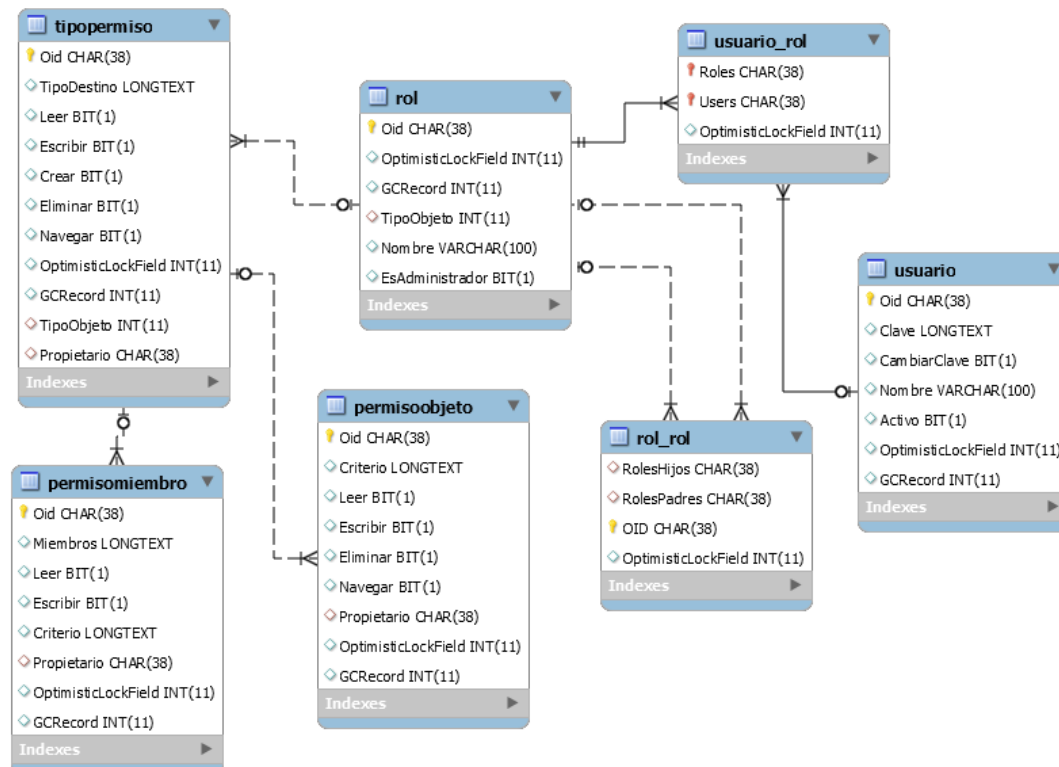
## 6.4 Modelo de pensum



## 6.5 Modelo de horarios



## 6.6 Modelo de usuarios

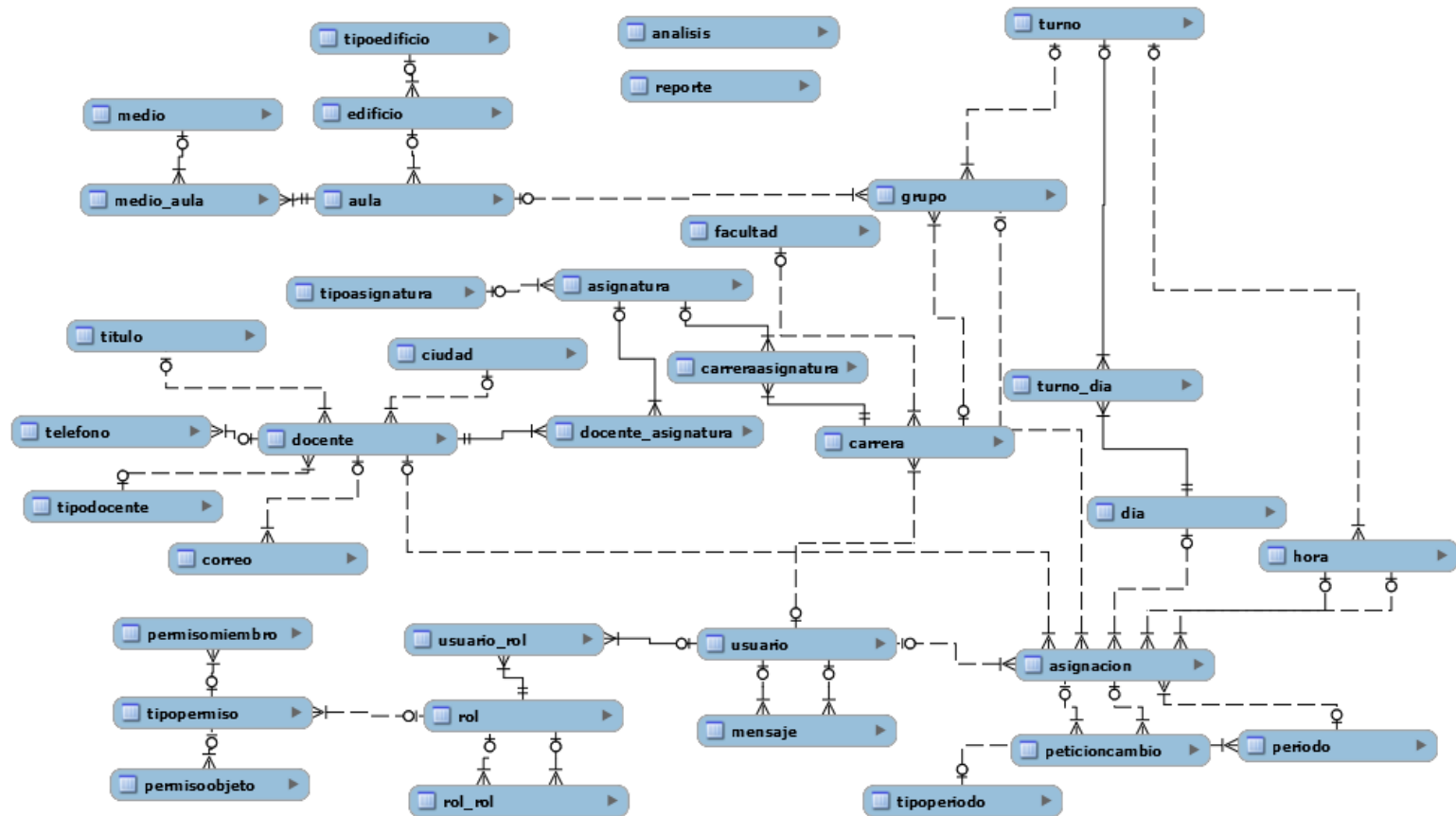


## 6.7 Modelo de reportes

reporte
⚡ Old CHAR(38)
◊ TipoObjeto TEXT
◊ Contenido LONGBLOB
◊ Nombre VARCHAR(100)
◊ Parametros TEXT
◊ TipoPredefinido TEXT
◊ OptimisticLockField INT(11)
◊ GCRecord INT(11)
Indexes ▶

analisis
⚡ Old CHAR(38)
◊ CadenaPropiedades LONGTEXT
◊ Nombre VARCHAR(100)
◊ Criterio LONGTEXT
◊ TipoObjeto VARCHAR(100)
◊ ContenidoGrafico LONGBLOB
◊ ContenidoTabla LONGBLOB
◊ OptimisticLockField INT(11)
◊ GCRecord INT(11)
Indexes ▶

## 6.8 Modelo global



## 6.9 Diccionario de datos

<b>PK</b>	Clave primaria.
<b>NN</b>	No nulo.
<b>UQ</b>	Único.
<b>BIN</b>	Binario.
<b>UN</b>	Sin signo.
<b>ZF</b>	Rellenar con cero.
<b>AI</b>	Autoincremental.
<b>FK</b>	Clave foránea.

Análisis										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Oid	GUID	✓	✓							
CadenaPropiedades	LONGTEXT		✓							
Nombre	VARCHAR(100)		✓							
Criterio	LONGTEXT		✓							
TipoObjeto	VARCHAR(100)		✓							
ContenidoGrafico	LONGBLOB		✓		✓					
ContenidoTabla	LONGBLOB		✓		✓					
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Asignación										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Usuario	CHAR(38)		✓							FK
EstadoAsignacion	INT(11)		✓							
Grupo	INT(11)		✓							FK
Docente	INT(11)		✓							FK
Asignatura	INT(11)		✓							FK



## Sistema para Planificación de Horarios Académicos

Periodo	INT(11)		✓							FK
Dia	INT(11)		✓							FK
Fin	INT(11)		✓							FK
Inicio	INT(11)		✓							FK
Descripcion	VARCHAR(100)									
Etiqueta	INT(11)									
Titulo	VARCHAR(100)									
InfoRecurrente	LONGTEXT		✓							
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Asignatura										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaActivo	BIT(1)		✓						1	
TipoAsignatura	INT(11)		✓							FK
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Aula										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaActivo	BIT(1)		✓						1	
Capacidad	INT(11)		✓			✓				
Edificio	INT(11)		✓							FK
OptimisticLockField	INT(11)									
GRecord	INT(11)									

## Sistema para Planificación de Horarios Académicos

Carrera										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
Usuario	CHAR(38)		✓							FK
EstaActivo	BIT(1)		✓						1	
Facultad	INT(11)		✓							FK
Duracion	INT(11)		✓			✓			10	En semestres
OptimisticLockField	INT(11)									
GRecord	INT(11)									

CarreraAsignatura										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Carrera	INT(11)	✓	✓							FK
Asignatura	INT(11)	✓	✓							FK
Semestre	INT(11)		✓							
Duracion	INT(11)		✓			✓				En horas
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Ciudad										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
OptimisticLockField	INT(11)									
GRecord	INT(11)									

## Sistema para Planificación de Horarios Académicos

Correo										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
Docente	INT(11)		✓							FK
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Dia										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Docente										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaBaneado	BIT(1)		✓						0	
EstaActivo	BIT(1)		✓						1	
TipoDocente	INT(11)		✓							FK
Titulo	INT(11)									FK
Apellido	VARCHAR(100)		✓							
Direccion	LONGTEXT									
Ciudad	INT(11)									FK
OptimisticLockField	INT(11)									

## Sistema para Planificación de Horarios Académicos

GRecord	INT(11)									
---------	---------	--	--	--	--	--	--	--	--	--

Docente_Asignatura										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Asignaturas	INT(11)	✓	✓							FK
Docentes	INT(11)	✓	✓							FK
OptimisticLockField	INT(11)									

Edificio										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaActivo	BIT(1)		✓						1	
Capacidad	INT(11)		✓			✓				
TipoEdificio	INT(11)		✓							FK
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Facultad										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaActivo	BIT(1)		✓						1	
Siglas	VARCHAR(100)		✓							
OptimisticLockField	INT(11)									
GRecord	INT(11)									

## Sistema para Planificación de Horarios Académicos

Grupo											
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario	
OID	INT(11)	✓	✓					✓			
Nombre	VARCHAR(100)		✓								
Comentario	LONGTEXT										
Creado	DATE										
Turno	INT(11)		✓							FK	
Aula	INT(11)		✓							FK	
Carrera	INT(11)		✓							FK	
Nivel	INT(11)		✓			✓					
ALectivo	INT(11)		✓			✓					
OptimisticLockField	INT(11)										
GCRecord	INT(11)										

Hora											
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario	
OID	INT(11)	✓	✓					✓			
Nombre	VARCHAR(100)		✓								
Comentario	LONGTEXT										
Creado	DATE										
Turno	INT(11)		✓							FK	
Inicio	DOUBLE		✓							TimeSpan	
Fin	DOUBLE		✓							TimeSpan	
OptimisticLockField	INT(11)										
GCRecord	INT(11)										

Medio											
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario	
OID	INT(11)	✓	✓					✓			

## Sistema para Planificación de Horarios Académicos

Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Medio_Aula										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Aulas	INT(11)	✓	✓							FK
Medios	INT(11)	✓	✓							FK
OptimisticLockField	INT(11)									

Mensaje										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
Remitente	CHAR(38)		✓							FK
Destinatario	CHAR(38)		✓							FK
FueLeido	BIT(1)		✓						0	
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Periodo										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaActivo	BIT(1)		✓						1	
TipoPeriodo	INT(11)		✓							FK

## Sistema para Planificación de Horarios Académicos

ALectivo	INT(11)		✓			✓				
Posicion	INT(11)		✓			✓			1	
Inicio	DATE									
OptimisticLockField	INT(11)									
GRecord	INT(11)									

PermisoMiembro										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Oid	CHAR(38)	✓	✓							
Miembros	LONGTEXT		✓							
Leer	BIT(1)		✓						0	
Escribir	BIT(1)		✓						0	
Criterio	LONGTEXT									
Propietario	CHAR(38)		✓							FK
OptimisticLockField	INT(11)									
GRecord	INT(11)									

PermisoObjeto										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Oid	CHAR(38)	✓	✓							
Criterio	LONGTEXT		✓							
Leer	BIT(1)		✓						0	
Escribir	BIT(1)		✓						0	
Eliminar	BIT(1)		✓						0	
Navegar	BIT(1)		✓						0	
Propietario	CHAR(38)		✓							FK
OptimisticLockField	INT(11)									
GRecord	INT(11)									

## Sistema para Planificación de Horarios Académicos

PeticiónCambio										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Origen	INT(11)		✓							FK
Destino	INT(11)		✓							FK
Mensaje	LONGTEXT		✓							
EstadoCambio	INT(11)		✓							
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Reporte										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Oid	GUID	✓	✓							
TipoObjeto	TEXT		✓							
Contenido	LOB		✓		✓					
Nombre	VARCHAR(100)		✓							
Parametros	TEXT									
TipoPredefinido	TEXT									
OptimisticLockField	INT(11)									
GRecord	INT(11)									

Rol										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Oid	CHAR(38)	✓	✓							
OptimisticLockField	INT(11)									
GRecord	INT(11)									
TipoObjeto	INT(11)		✓							
Nombre	VARCHAR(100)		✓							
EsAdministrador	BIT(1)		✓						0	



## Sistema para Planificación de Horarios Académicos

Rol_Rol										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
RolesHijos	CHAR(38)		✓							FK
RolesPadres	CHAR(38)		✓							FK
OID	CHAR(38)	✓	✓					✓		
OptimisticLockField	INT(11)									

Telefono										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
TipoTelefono	INT(11)		✓						0	
Docente	INT(11)		✓							FK
OptimisticLockField	INT(11)									
GRecord	INT(11)									

TipoAsignatura										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
OptimisticLockField	INT(11)									
GRecord	INT(11)									

TipoDocente										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							

## Sistema para Planificación de Horarios Académicos

Comentario	LONGTEXT									
Creado	DATE									
OptimisticLockField	INT(11)									
GCRecord	INT(11)									

TipoEdificio										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
OptimisticLockField	INT(11)									
GCRecord	INT(11)									

TipoPeriodo										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaActivo	BIT(1)		✓						1	
Duracion	INT(11)		✓							
Semanas	INT(11)		✓							
OptimisticLockField	INT(11)									
GCRecord	INT(11)									

TipoPermiso										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Oid	CHAR(38)	✓	✓							
TipoDestino	LONGTEXT		✓							
Leer	BIT(1)		✓						0	
Escribir	BIT(1)		✓						0	

## Sistema para Planificación de Horarios Académicos

Crear	BIT(1)		✓						0	
Eliminar	BIT(1)		✓						0	
Navegar	BIT(1)		✓						0	
OptimisticLockField	INT(11)									
GCRecord	INT(11)									
TipoObjeto	INT(11)		✓							
Propietario	CHAR(38)		✓							FK

Titulo										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
OptimisticLockField	INT(11)									
GCRecord	INT(11)									

Turno										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
OID	INT(11)	✓	✓					✓		
Nombre	VARCHAR(100)		✓							
Comentario	LONGTEXT									
Creado	DATE									
EstaActivo	BIT(1)		✓						1	
OptimisticLockField	INT(11)									
GCRecord	INT(11)									

Turno_Dia										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Dias	INT(11)	✓	✓							FK
Turnos	INT(11)	✓	✓							FK
OptimisticLockField	INT(11)									

## Sistema para Planificación de Horarios Académicos

Usuario										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Oid	CHAR(38)	✓	✓							
Clave	LONGTEXT		✓		✓					
CambiarClave	BIT(1)		✓						1	
Nombre	VARCHAR(100)		✓							
Activo	BIT(1)		✓						1	
OptimisticLockField	INT(11)									
GCRecord	INT(11)									

Usuario_Rol										
Nombre	Tipo	PK	NN	UQ	BIN	UN	ZF	AI	Por defecto	Comentario
Roles	CHAR(38)	✓	✓							FK
Users	CHAR(38)	✓	✓							FK
OptimisticLockField	INT(11)									

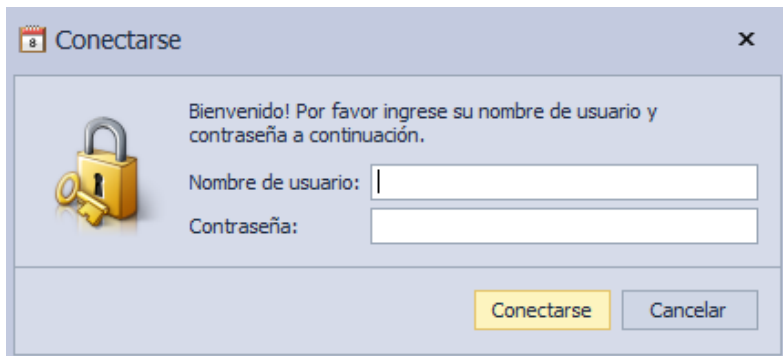
## Capítulo 7. Presentación del sistema

Producto de este trabajo se desarrolló el denominado ***Sistema para Planificación de Horarios Académicos (SIPHA)***, con el propósito de servir como herramienta para agilizar el proceso de creación de horarios en el Recinto Universitario Augusto C. Sandino.

SIPHA fue desarrollado en el lenguaje de programación VB .Net, utilizando el NetFramework 4.0, esto con el objetivo de mantener compatibilidad con el sistema operativo Windows XP. Trabaja además con una arquitectura cliente-servidor, conectándose al gestor de base de datos MySQL Server.

### 7.1 Inicio de sesión

El formulario de inicio de sesión está formado por dos cajas de texto, una para el nombre de usuario y otra para la clave secreta, una vez ingresados estos datos, debe presionar en el botón “Conectarse” para ingresar a la pantalla principal, si desea salir del sistema, presione el botón “Cancelar”



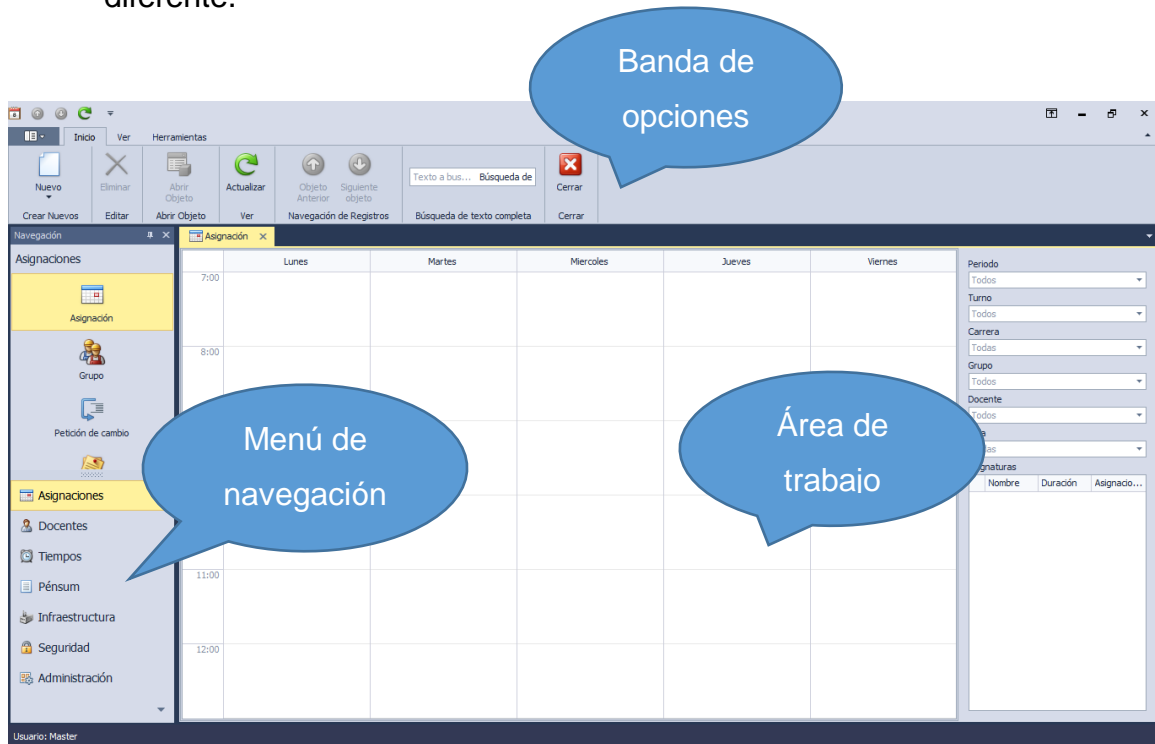
### 7.2 Pantalla principal

La pantalla de principal está formada por 3 áreas principales:

- En el lateral izquierdo, se encuentra el menú de navegación, el cual está formado a su vez por dos paneles. En la parte inferior, deberá seleccionar el grupo de módulo a utilizar, y en la parte superior el módulo específico.
- En la parte superior se encuentra la barra de opciones, en este lugar

aparecen las acciones que puede ejecutar en cada ventana del sistema, por ejemplo: nuevo, guardar y refrescar.

- En la parte central se encuentra el área de trabajo, acá aparecerán los controles disponibles para trabajar, cada ventana tiene un área de trabajo diferente.



### 7.3 Control de usuarios

La ventana de control de usuarios proporciona un lugar para crear y administrar los usuarios que tendrán acceso al sistema, originalmente solo la cuenta de administrador tiene autorización de ingresar a este módulo, sin embargo, el administrador puede asignar privilegios a otra cuenta para realizar los trabajos de asignación de permisos y creación de usuarios.

Para crear una cuenta, es necesario proporcionar un nombre de usuario y al menos un rol asociado.

Nombre de usuario: Master

☒ Esta activo

☐ Cambiar clave al iniciar sesión

Roles

Nombre	Es adminis...	Puede edit...
Administrador	<input checked="" type="checkbox"/>	<input type="checkbox"/>

El formulario de administración de roles proporciona un lugar para definir grupos de permisos que posteriormente podrán ser asignados a usuarios individuales. Cada cuenta puede tener varios roles asociados, y por cada rol se especifica el detalle de permiso asignado. Los permisos pueden establecerse a nivel general, por miembros (campos) y por objetos (registros).

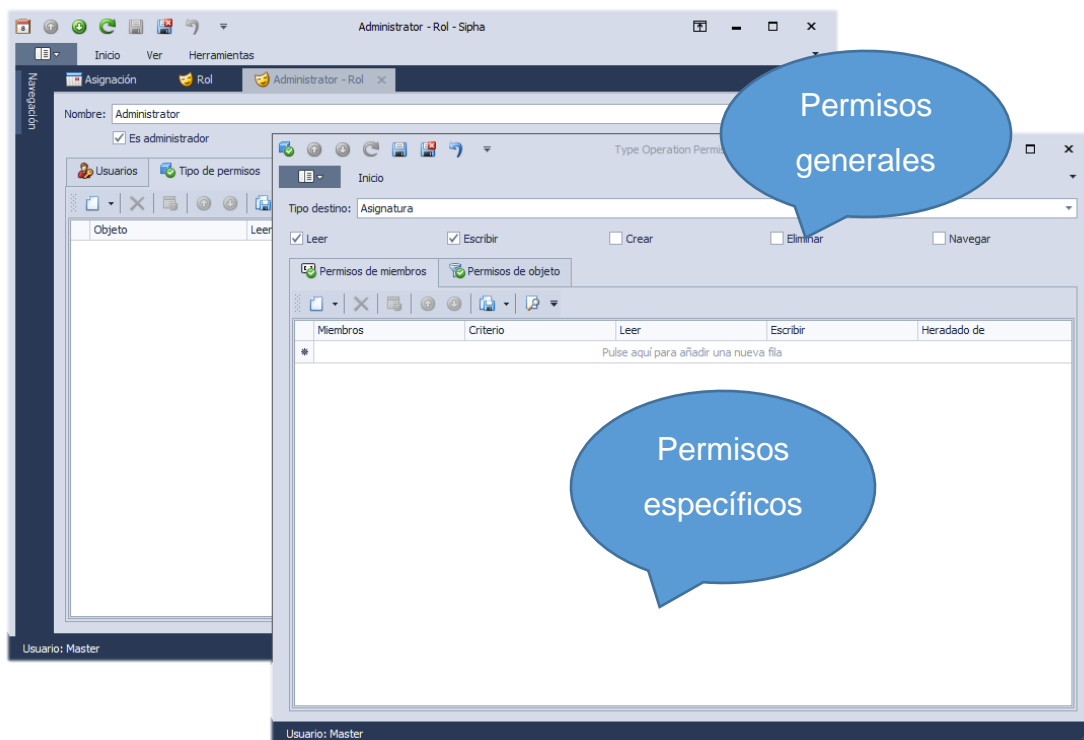
Para los permisos generales se debe seleccionar el módulo, y marcar las opciones deseadas (Leer, Escribir, Crear, Navegar, Eliminar)

- Leer: Permite la visualización de registros.
- Escribir: Permite la modificación de registros existentes.
- Crear: Permite la creación de nuevos registros
- Navegar: Permite la visualización del módulo en el menú de navegación.
- Eliminar: Permite la eliminación de registros. Este permiso es de uso restringido, ya que aunque se marque esta casilla, si un módulo no admite

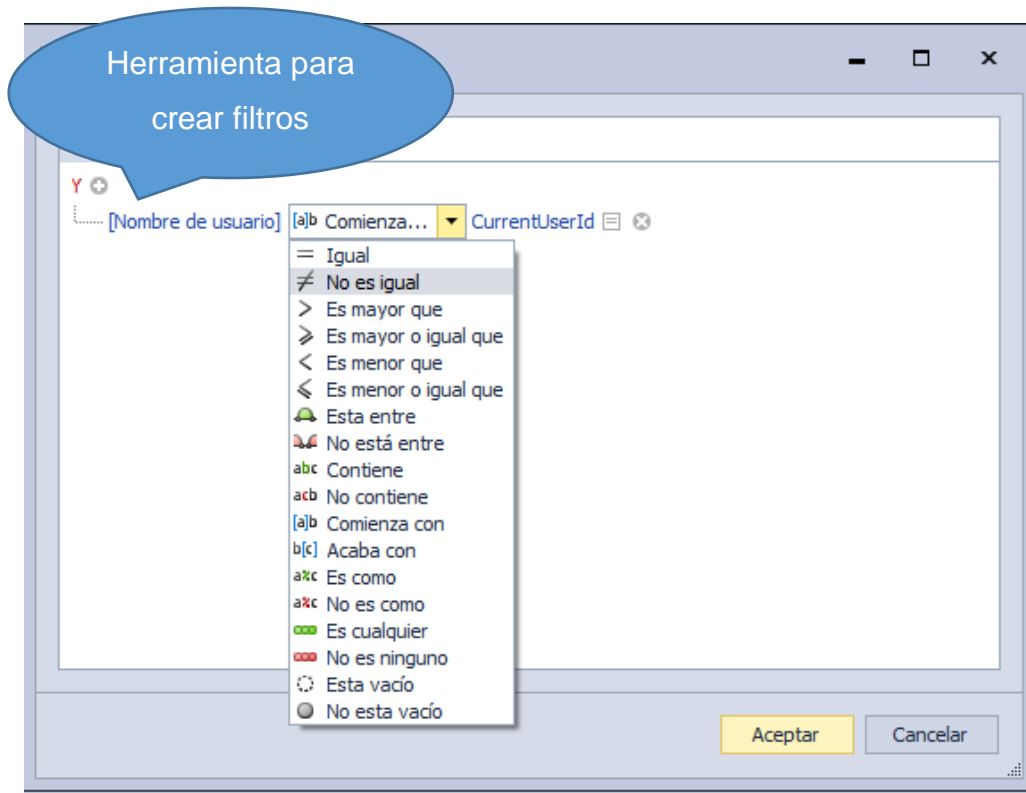
la eliminación de registros, la opción “*Eliminar*” no aparecerá en la banda de opciones.

Los permisos miembros representan permisos específicos para cada campo, se deben crear dentro de los permisos generales, e indicar el nombre de los campos que desea desbloquear (seleccionandolos de una lista).

Los permisos de objeto representarn permisos para registros específicos, se deben crear dentro de los permisos generales, e indicar el criterio o filtro de los registros que desea desbloquear (docentes cuyo titulo sea “*Ingeniero*”, por ejemplo). Para facilitar este proceso, se proporciona de una herramienta para crear filtros de manera visual.







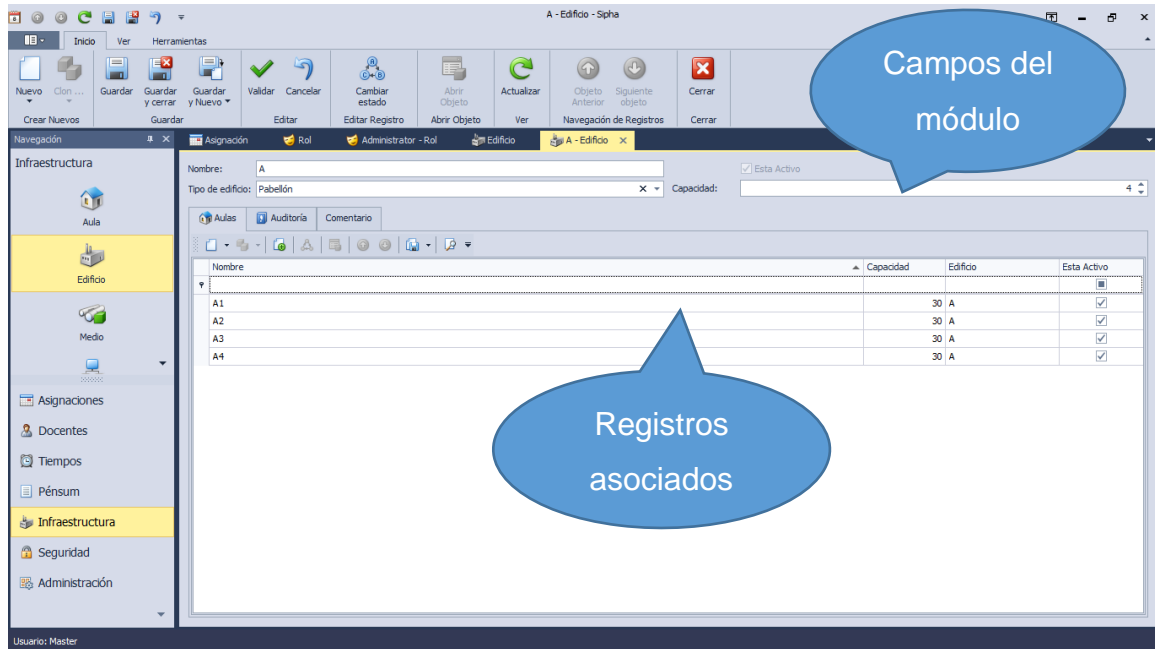
## 7.4 Crear registros

Los formularios de creación de registros son diferentes para cada módulo, sin embargo, el proceso es igual para todo el sistema. Una vez seleccionado el modulo con el que se desea trabajar, se deben llenar los campos solicitados, y posteriormente presionar el botón guardar.

Si el módulo tiene registros asociados, por ejemplo: los pabellones tienen aulas, puede crearlos en la misma venta, haciendo clic en la tabla correspondiente de registros asociados.

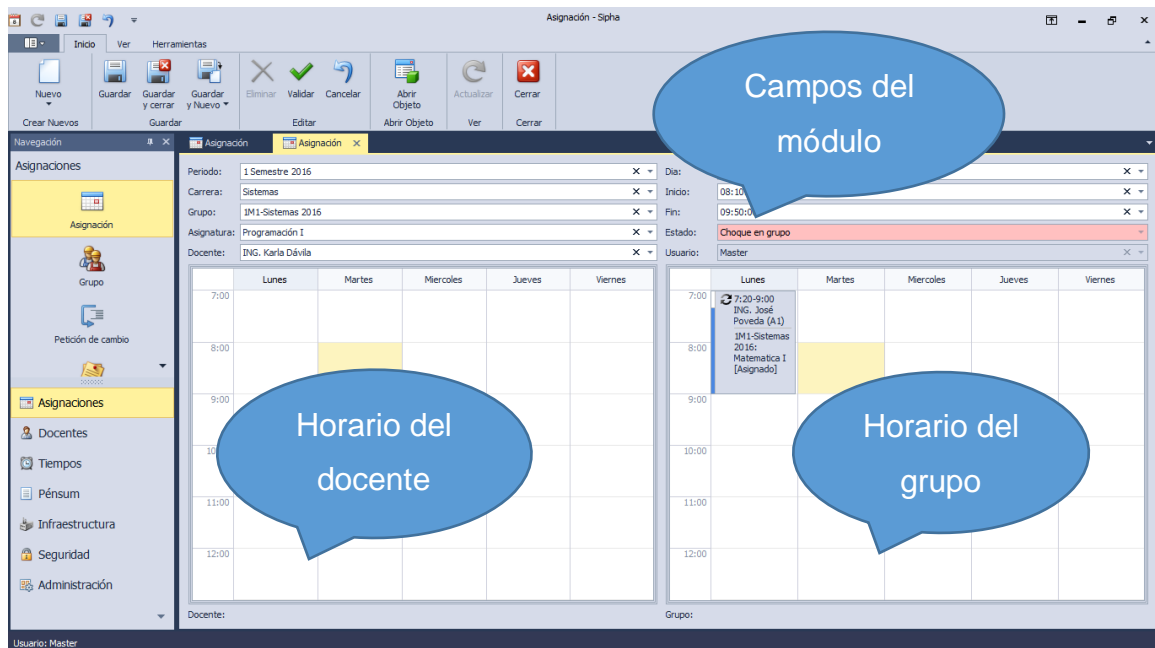
La mayoría de los registros permiten el cambio de estado entre “*Activo*” e “*Inactivo*”, los registros inactivos no son tomados en cuenta en la creación de nuevos registros. Para alternar entre estos dos estados presione el botón “*Cambiar estado*”.

## Sistema para Planificación de Horarios Académicos



### 7.5 Formar horarios

El proceso de creación de horarios es similar a la creación de los demás registros, la diferencia radica en que mientras llena los campos solicitados podrá visualizar el horario actual del docente y del grupo seleccionado, además que verá el estado de la asignación en tiempo real.



## 7.6 Listar registros

Mientras navega entre los módulos podrá visualizar los registros almacenados en una tabla con las columnas más relevantes, para acceder a un registro particular, solamente debe presionar doble clic sobre él.

Las listas de registros se muestran en tablas, a excepción del módulo de asignaciones que los muestra en forma de horario (semanal, diario, mensual, según como esté configurado el turno seleccionado).



Título	Esta Bar...	Nombre	Apellido	Esta Activo	Tipo de docente	Comentario	Ciudad
ING.	<input type="checkbox"/>	José	Poveda	<input checked="" type="checkbox"/>	Planta		Matagalpa
ING.	<input type="checkbox"/>	Karla	Dávila	<input checked="" type="checkbox"/>	Planta		Estel
LIC.	<input type="checkbox"/>	Victor	Parrales	<input type="checkbox"/>	Horario		Managua



	Lunes	Martes	Miércoles	Jueves	Viernes
7:00	7:20-9:00 ING. José Poveda (A.1)				
8:00	IM1-Sistemas 2016: Matemática I [Asignado]				
9:00					
10:00			10:50-12:30 ING. Karla Dávila (A.1)		
11:00			IM1-Sistemas 2016: Programación I [Asignado]		
12:00					

## 7.7 Filtrar horarios

Nombre	Duración	Asignacio...
Matemat...	56	18
Program...	56	0

La lista de asignaciones muestra inicialmente el horario de un único grupo, si desea filtrar los horarios mostrados, debe hacer uso del control ubicado en la parte lateral derecha, en él podrá especificar, el periodo, turno, carrera, grupo, docente y aula que desea mostrar.

El uso de estos filtros es opcional, de modo que puede aplicar solamente algunos de ellos, o todos, según convenga.

Se muestra además en este control, la lista de asignatura que cursa el grupo seleccionado, junto al total de horas esperado (según la configuración del pensum) y el total de horas asignadas actualmente.

## 7.8 Reportes

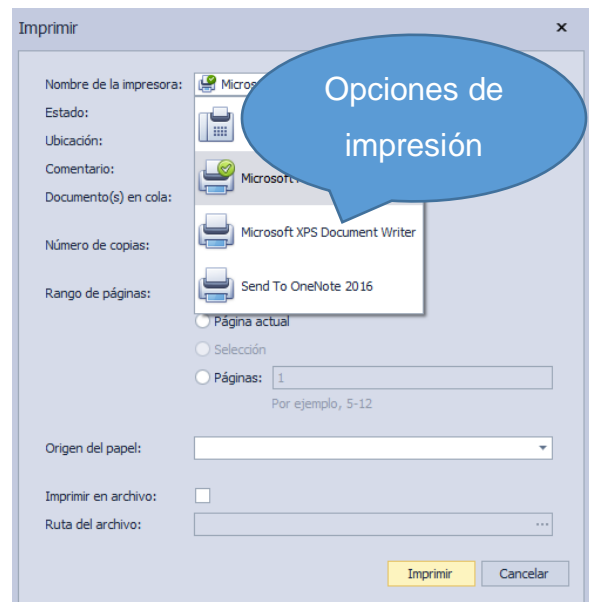
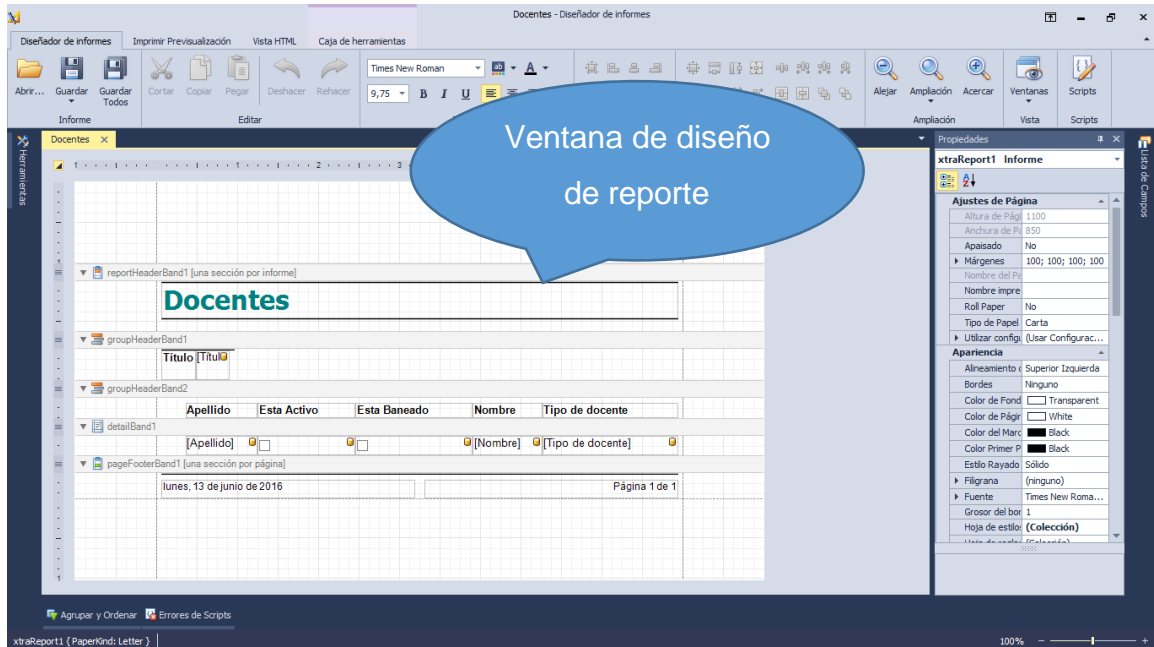
El módulo de reportes permite crear informes de cualquier conjunto de datos almacenados en el sistema, guardar el formato y posteriormente ejecutar el reporte para extraer la información en el momento deseado, sin volver a configurar los formatos en cada ejecución.

Para crear un reporte deberá usar el asistente de creación de reportes, en el cual se le pedirá especificar principalmente el módulo a usar y los campos que desea incluir, opcionalmente puede seleccionar también configuraciones de formato y apariencia.

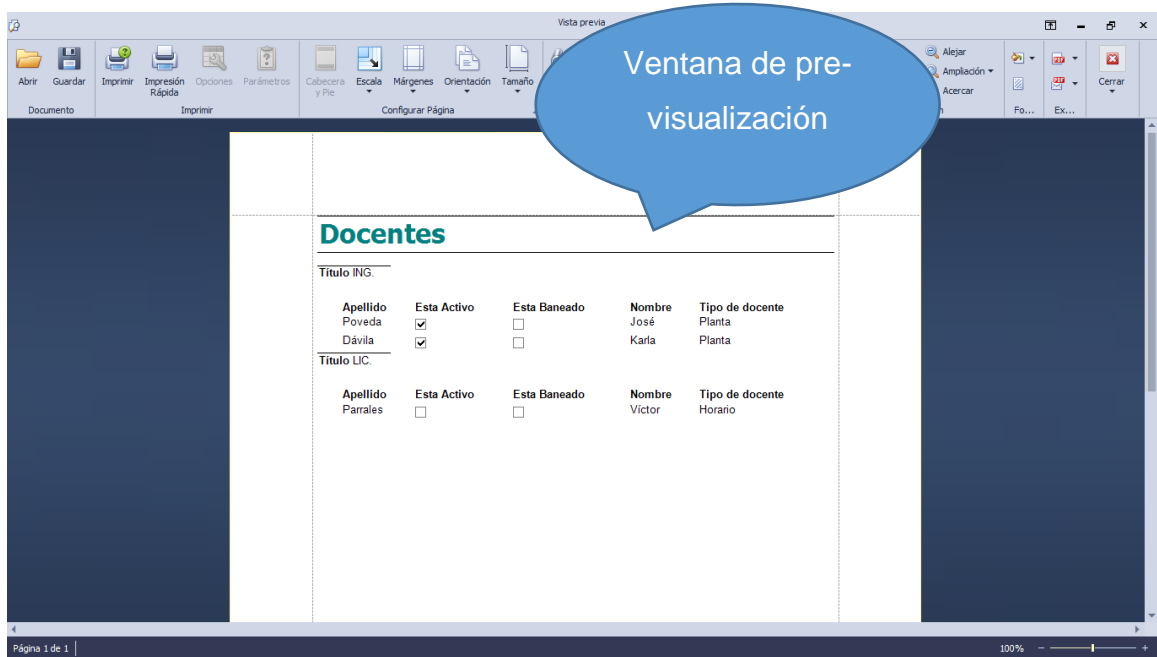
Una vez finalizado el asistente se mostrará la ventana de diseño, en la que puede personalizar aún más el reporte, en esta ventana puede incluir texto estático, filtros, campos enlazados, imágenes, tablas, formas, formatos, colores, entre otras opciones.

## Sistema para Planificación de Horarios Académicos

Para ejecutar un reporte creado previamente debe presionar doble clic sobre el formato deseado en la tabla de registros de reporte. Se mostrará una ventana de pre-visualización en la que podrá enviar el archivo a una impresora física, una impresora virtual, a un archivo, o enviarlo directamente por correo electrónico. En caso de usar un archivo o un correo electrónico como destino, tendrá a su disposición los siguientes formatos: pdf, xls, xlsx, mht, txt, jpg, csv, rtf y html.



## Sistema para Planificación de Horarios Académicos



### 7.8.1 Análisis

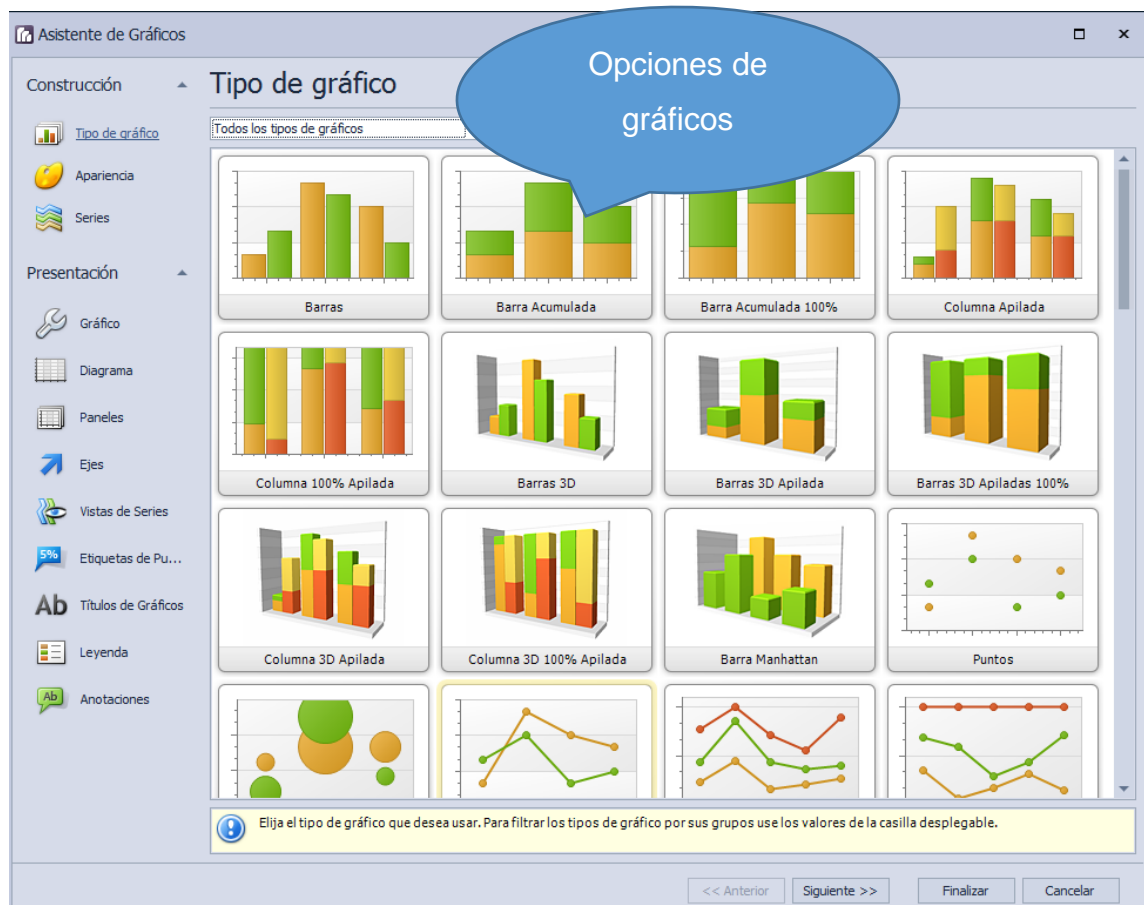
Además de los reportes clásicos, el sistema propuesto cuenta con un módulo de análisis en el que puede usar tablas dinámicas y gráficos para resumir y presentar la información almacenada.

Para el uso de las tablas, deberá seleccionar el modulo que desea analizar y seleccionar los campos que usará como columnas, filas y datos. Los gráficos se generan a partir de la tabla dinámica, sus opciones se centran en el tipo de gráfico y la apariencia del mismo.

Tabla de resumen

Docentes			
Pivote			
Esta Baneado	Tipo de docente	Apellido	Direccion
Nombre	Comentario	Oid	Título ▲
Ciudad ▲	ING.	LIC.	Total General
Estelí	1		1
Managua		1	1
Matagalpa	1		1
Total General	2	1	3

## Sistema para Planificación de Horarios Académicos



## Capítulo 8. Pruebas en entorno controlado

### 8.1 Introducción

Proyecto:		Tipo de proyecto	
Sistema para planificación de horarios académicos (SIPHA)		Proyecto de desarrollo de software a la medida.	
Documentos de evaluación relacionados			
Requerimientos.pdf			
Diseños.pdf			
Equipo de proyecto			
Jefe de equipo	Sergio A. Velásquez	Arquitecto de producto	Sergio A. Velásquez Justo L. Rodríguez

### 8.2 Objetivos del plan

Este documento, tiene como finalidad entregar las pautas y definir la estrategia que se seguirá para llevar a cabo la validación del software SIPHA.

El objetivo general del plan es establecer la cronología y condiciones para la aplicación de las pruebas de manera de obtener, un sistema que pueda ser completado con una recepción total de los interesados y entrar en operación con la totalidad de las funcionalidades requeridas para su funcionamiento.

### 8.3 Documentos relacionados

Nombre	Descripción
Requerimientos.pdf	Informe sobre requerimientos funcionales y no funcionales.
Diseños.pdf	Informe con diseños UML y Diccionario de datos del sistema.



#### 8.4 Alcance de las pruebas

Módulos del sistema a ser probados:	<ul style="list-style-type: none"> <li>• Parámetros del sistema</li> <li>• Gestión de horarios</li> <li>• Gestión de usuarios</li> <li>• Reportes</li> </ul>
Objetivos de las pruebas	<p>En estos Módulos se realizarán pruebas para validar:</p> <ul style="list-style-type: none"> <li>• La visualización de los datos, ingresados o modificados.</li> <li>• La respuesta y realización de las transacciones de cada módulo.</li> <li>• Que los estados de las actividades generadas en el sistema se reflejen de acuerdo a la secuencia lógica requerida por el usuario.</li> <li>• La secuencia lógica de las funcionalidades y transacciones.</li> </ul>
Detalle del orden de ejecución de los módulos	<p>Los módulos se deben ejecutar en forma independiente, pero consecutivos en el orden siguiente:</p> <ul style="list-style-type: none"> <li>• Gestión de usuarios.</li> <li>• Parámetros del sistema.</li> <li>• Gestión de horarios.</li> <li>• Reportes.</li> </ul>
Responsabilidad de la prueba	<p>Las pruebas serán responsabilidad del equipo de desarrollo, quienes en conjunto con los usuarios deben seleccionar y ejecutar las pruebas que aseguren la efectividad del sistema.</p>

## 8.5 Entorno y configuración de las pruebas

Para el proceso de pruebas del proyecto se requiere de la disponibilidad de los siguientes entornos:

- Servidor Windows Server/Linux con MySQL 5.5 o superior, conexión de red, permitidas las conexiones LAN al servicio MySQL a través del puerto 3306.
- Equipos Cliente: PC Windows XP o superior, conexión de red (en LAN con el servidor) y .Net Framework 4.0.

Todos ellos configurados por el equipo de desarrollo.

## 8.6 Criterios de aprobación o rechazo

Se aprobará el proyecto con un 100% de las pruebas ejecutadas, pero con un 90% de aceptación. Esto quiere decir el 90% de las pruebas deben ser exitosas y sin errores. El restante 10% pueden existir errores medios o bajos, pero no graves.

En caso de ocurrir que el proyecto no cumpla con el nivel exigido, el proyecto deberá ser ajustado y sometido nuevamente al proceso de prueba.

### 8.6.1 Tipos de errores

<b>Errores graves</b>	Información crítica presentada erróneamente, información mal registrada en la base de datos, caídas de programas o incumplimiento de objetivos en funciones principales.
<b>Errores medios</b>	Errores en documentos impresos que se entregan a personas ajenas a la organización, errores en presentación de datos, incumplimiento de objetivos en funciones secundarias o caídas de programas auxiliares.
<b>Errores leves</b>	Errores en presentación de datos secundarios, no adecuación a estándares, comportamientos correctos pero diferentes en situaciones similares o dificultades de operación.

### **8.7 Estrategia de pruebas**

Se requiere validar por parte del equipo de desarrollo y por parte del usuario al producto en tres etapas:

- 1ra. Etapa: Que los permisos de usuarios funcionen correctamente, restringiendo el uso y acceso al sistema.
- 2da. Etapa: Que la funcionalidad relacionada a la definición de parámetros funcione correctamente, almacenando y recuperando información de manera precisa y eficiente.
- 3ra. Etapa: Que la funcionalidad relacionada a la formación de horarios funcione correctamente, mostrando los choques entre asignaciones y filtrando la información entregada al usuario.

### **8.8 Escenario de las pruebas**

Para cumplir con los objetivos planteados deben existir tres escenarios, que son, pruebas de instalación, pruebas de interfaz gráfica y pruebas funcionales.

Para las Pruebas de Instalación se debe comprobar que:

- La aplicación no presenta anomalías, como cuelgues al arrancar, o errores por dll faltante.
- Que apunta al servidor y base de datos definidos.

Para las pruebas de GUI se debe comprobar que:

- Comportamiento de aplicación con casos de bordes inválidos y válidos, donde las pruebas de borde se definen como aquellas pruebas en las cuáles los datos de prueba a utilizar son valores límites.
- Carga, despliegue, foco, modalidad, navegabilidad y usabilidad de las GUI del sistema y sus elementos.

Para las pruebas funcionales se debe comprobar:

- Se cumple con los requerimientos funcionales definidos.

## Sistema para Planificación de Horarios Académicos

- El sistema notifica inmediatamente en casos de choque de horarios.
- El comportamiento de aplicación con casos inválidos y válidos, de flujo completo del proceso de creación de horarios.

## V. Conclusiones

La creación de horarios académicos es una actividad fuertemente ligada al proceso educativo que llevan a cabo las instituciones de educación superior, es por ello que su correcta gestión es de vital importancia. En un entorno cambiante, donde la complejidad del proceso de creación de horarios aumenta cada año, el uso de herramientas que simplifiquen esta tarea y garanticen el cumplimiento de un conjunto mínimo de reglas, se convierte en una necesidad.

El Recinto Universitario Augusto C. Sandio desde su fundación ha mostrado un incremento en el número de grupos, aulas y docentes involucrados en el proceso de formación de horarios, por consiguiente, un aumento en la dificultad de formar horarios.

Para reducir de manera efectiva las dificultades existentes en la creación de horarios, el sistema a implementar debe cumplir con la totalidad de los requerimientos definidos en el Capítulo 3 del presente trabajo. Estos requerimientos se han obtenidos a partir del análisis del procedimiento actual, la observación y las sugerencias de los involucrados.

La creación e implementación del “Sistema para planificación de horarios académicos” es completamente factible en el Recinto Universitario Augusto C. Sandio, ya que este cuenta con los recursos, técnicos y humanos necesarios para el correcto funcionamiento del sistema propuesto.

Dado que el sistema elaborado ha superado exitosamente las pruebas propuestas, incluidas las pruebas funcionales, y permite formar horarios académicos basado en los requerimientos planteados por las autoridades de la sede, podemos concluir que los resultados del presente trabajo son satisfactorios y están alineados con los objetivos propuestos inicialmente.

## VI. Recomendaciones

- Los parámetros correspondientes a datos bases, tales como: aulas, turnos y docentes, deben verificarse y actualizarse al menos una vez al año. Estos parámetros bases sirven como bloques en la construcción de los horarios académicos, la precisión de los mismos repercute directamente en el resultado final generado por el sistema.
- Crear cuentas de usuario con roles de “Docente”, e instalar la aplicación en las computadoras de la “Sala de docentes” con el fin de proporcionar a los maestros la posibilidad de consultar sus horarios y enviar comentarios a los coordinadores antes que sean publicados de manera oficial.
- En caso que un recurso deje de estar disponible, deberá marcarse como inactivo, bajo ninguna circunstancia deberá intentarse la eliminación total de los registros. En caso de actualización de planes de estudio, deberán configurarse en el módulo de pensum, desactivando el plan anterior y guardando el plan actualizado.
- Realizar capacitaciones para los involucrados. Aunque los coordinadores y docentes tienen considerables conocimientos informáticos y son capacitados constantemente en el manejo de herramientas digitales, es recomendable involucrarlos en el manejo del sistema propuesto, tanto para obtener retroalimentación como para garantizar un máximo aprovechamiento de las capacidades del sistema.
- Los módulos de reporte y análisis pueden usarse para generar estadística a partir de los datos históricos, se debe alimentar el sistema con datos precisos, para posteriormente extraer información resumida.

## VII. Bibliografía

Angell, I. a. (2001). *Information Systems Management: Opportunities and Risks*.

Beck, K. (2001). *Manifesto for Agile Software Development*. Agile Alliance.

Cockburn, A. (2000). *Agile Software Development*. Highsmith Series.

Jimenez, A. (2013). *Diagnóstico en educación: Modelo, Técnicas e Instrumentos*.  
Salamanca: Amaru Ediciones.

Kendall, K., & Kendall, J. (2005). *Análisis y diseño de sistemas*. Mexico: Pearson Educacion.

Laudon, K., & Laudon, J. (2012). *Sistemas de información gerencial- Administración de la empresa digital*. México: Pearson Educación- Prentice Hall.

Microsoft. (20 de Julio de 2015). VS 2015. Obtenido de Visual Studio Web site:  
<https://www.visualstudio.com/news/vs2015-vs>

Oracle Corporation. (2014). *Why MySQL*. Obtenido de MySQL:  
<http://www.mysql.com/why-mysql/>

Palacio, J. (2007). *Flexibilidad con Scrum*.

Real Academia Española. (2005). *Diccionario de la lengua española*. Madrid: Espasa Calpe.

Schmuller, J. (2001). *Aprendiendo UML*. México: Prentice Hall.

Schwaber, K. (2010). *Advanced Development Methods. SCRUM Development Process*. Burlington.

SENN, J. A. (1992). *Análisis y Diseño de Sistemas de Información*. Mexico: McGrawHill.

Silberschatz, A. (2002). *Fundamentos de bases de datos*. Madrid: McGraw Hill.

Sosa Sállico, M. (2006). *Sistema Organizacional. Estructura Organizacional. Conceptos Afines*.

Sosa Sállico, M. (2007). *Estructura organizacional*.

Tanenbaum, A. S. (2009). *Sistemas operativos modernos (3 edición)*. Prentice Hall.



## **VIII. Anexos**

### **A. Entrevista a coordinadores de carrera**

Objetivo:

- Obtener los requerimientos funcionales que deberá cumplir el sistema propuesto.
  - Determinar los roles involucrados en el proceso y sus respectivas responsabilidades.
- 
1. ¿Quiénes son los involucrados en el proceso de creación de horarios académicos?
  2. Para cada uno de los involucrados, mencione:
    - a. Funciones que realiza
    - b. Datos que aporta (entradas)
    - c. Información que solicita (salidas)
  3. Explique el proceso actual de creación de horarios académicos.
  4. ¿Qué formatos se utilizan en el proceso y con qué frecuencia?
  5. ¿Qué funciones extras cree conveniente incluir en el software a desarrollar?
  6. Existen documentos oficiales que normen el proceso de creación de horarios.
    - a. En caso afirmativo liste los documentos.